IEAST User Manual

IEAST - An Integrated Environment for Analyzing STARMA Models

by

Cheng-Yu Lee

Department of Forestry
Michigan State University
Copyright ©2004 February 14, 2005

Contents

Contents i

Preface 1

1 Introduction 3

1.1 What is IEAST . . . 3
1.2 Quick Start - A Simple Example . . . 4
1.3 Features . . . 13
1.4 Requirements . . . 13
1.5 Installation . . . 14
1.6 Performance . . . 14
1.7 Menu System . . . 14
1.8 Conventions . . . 16

2 STARMA Models and Modeling 17

2.1 Statistical Modeling . . . 17
2.2 Box-Jenkins Modeling . . . 18
2.3 STARMA Model Definition . . . 18
2.4 Elementary Statistical Tools for Analyzing STARMA Models . . . 20
2.5 Assumptions . . . 21
2.6 General STARMA Modeling Procedure for using IEAST . . . 21

3 Data Preparation 23

3.1 Load space-time dataset from .dat files . . . 23
3.2 Import space-time datasets from other formats . . . 24
3.3 Simulation Data . . . 27
3.4 Dataset from Excel files and other sources . . . 29

3.5 Important Global Variables 29
3.6 Visualize Your Dataset 30

4 Spatial Correlation Structure **33**
4.1 Spatial Dimension, N_SITE and N x N 33
4.2 Spatial Order Definition, SOD 33
4.3 Spatial Relation Matrix, SRM 34
4.4 Spatial Weighting Matrices, SWM 35
4.5 Configure Your System 35
4.6 Load Spatial Correlation Structure or Weight Matrices Directly from Files 37

5 Data Preprocessing **39**
5.1 Remove Mean 39
5.2 De-Seasonalize 39
5.3 Difference by One 41
5.4 De-Trend 41
5.5 Box-Cox Transform 44
5.6 Subsequencing/Resampling 45
5.7 Smoothing 45
5.8 Missing Data 46
5.9 Filter with a given STARMA model 47

6 Correlation Analyses **49**
6.1 The Role of STACF/STPACF in STARMA Modeling 49
6.2 Autocorrelations: STACF and STPACF 50
6.3 Cross Correlations: STXCF, STPXCF, and ExtSTXCF 53
6.4 Plotting correlations versus T-Lag/S-Lag for **Z/Y** 57

7 Model Identification **61**
7.1 Automatic Identification 61
7.2 Artificial Identification 62
7.3 Parameter Masking 62

8 Parameter Estimation **65**
8.1 Estimation of Model Parameters 65
8.2 Estimation of Spatial Weighting Structure 65

8.3 Estimation Algorithms in IEAST 66

9 Diagnostic Analysis **71**

9.1 STACF of Residuals 71

9.2 STPACF of Residuals 72

9.3 Statistical Significance 73

9.4 AICC/BIC Analysis 74

9.5 Residuals' Variance 76

10 Interpreter **77**

10.1 The Programming Environment 77

10.1.1 Invoke and Quit 77

10.1.2 Program Mode and Interactive Mode 78

10.1.3 Notices for the Interpreter 79

10.2 A Sample Program 79

10.2.1 Dataset Generation 80

10.2.2 Modeling Procedures 80

10.2.3 Program Description 81

10.2.4 Results 81

10.3 Command References 86

Bibliography **95**

List of Figures

1.1 Inputs and outputs of IEAST. 4

1.2 Space-time ACF and PACF for STAR(2,2) 10

1.3 The first two levels of the menu system in IEAST 15

2.1 Relationship between data-generating and statistical modeling procedures 17

2.2 Box-Jenkins Modeling 18

2.3 An example of definition of spatial orders in 2D systems 19

2.4 An extended STARMA modeling procedure. The area enclosed by dotted line is for the case that model type cannot be decided simply from autocorrelations. AICC/BIC of every possible model type is evaluated and compared to select the best model type. 22

3.1 An example of 3D plotting of dataset. 31

5.1 3D plots of the original dataset. 42

5.2 5th order polynomial spatial trend and temporal trend (average over space). 44

5.3 3D plots of the dataset after de-trending. 44

5.4 Dataset before (left) and after (right) IDM (d^2) smoothing with 3x3 window. . . . 46

5.5 Autocorrelation functions of a dataset before (left) and after (right) filtering. . . . 48

6.1 STACF and STPACF for STAR(2,1) 50

6.2 STACF and STPACF for STMA(1,1) 50

6.3 Extended cross correlations (ExtSTXCF) between two space-time variables Z and Y. 57

6.4 The relationship between space-time variable **Z** and independent variable **Y** under the assumption of space-time regression model. 57

6.5 STACF plots versus T-Lag (left) and S-Lag (right) 59

9.1 Residuals' STACF and STPACF for the example in Quick Start 72

10.1 STACF and STPACF for the simulated dataset STARMA(1,1) 81

List of Tables

1.1 Accuracies of model type selection using variances, AICC, BIC, and -AICC*BIC based on 150 Monte Carlo simulated datasets. 16

3.1 An example for format XY+T of SPACE:10x10 and TIME:1..28 24

3.2 An example for format T+XY of SPACE:10x10 and TIME:1..28 25

3.3 An example for format XYTD of SPACE:10x10 and TIME:1..28 25

Preface

Since the theoretical basis of space-time model established in 1970, there have never implemented any general-purpose software for space-time modeling. We began the development of IEAST, a true space-time analyzing environment, in the early 2000 by a grant supported by Michigan Agricultural Experiment Station (MAES), Michigan State University.

All work been done since then is in an *MatLab*®-like language called *GNU Octave*. As users may notice, the environment is a text-based menu selection. Combining with *GNU Plot*, IEAST provides a complete set of functionalities and sufficient 2D/3D graphical abilities for most space-time modeling scenarios. Several space-time modeling and analyses on spreading of various emerging infectious diseases among animals or human have been done successfully by using IEAST. The results were encouraging and promising, and we'd like to make IEAST be applicable to even more diverse disciplines. We keep improving IEAST both on its functionalities supported and applicability to various applications. Hopefully, in the future, there is enough funding or time to let us port the entire system to Window-based environment to further provide flexibility and being user-friendly.

The manual gives comprehensive instructions, examples of how to use IEAST, and some techniques in space-time modeling. Comments concerning errors or omissions in this edition will be greatly appreciated, so that they may be rectified in future editions.

Chapter 1

Introduction

1.1 What is IEAST

Real world datasets, in many disciplines, are often organized by units of time as well as by geographic locations. The underlying processes of these datasets are correlated not only in time but also in space. It is not always reasonable to analyze these stochastic processes by considering space and time separately or by using the well-developed univariate AutoRegressive Moving Average (ARMA) theory. Statistical modeling of the datasets produced by stochastic processes that are correlated in both space and time results in the space-time extension of univariate ARMA time series models, i.e. Space-Time AutoRegressive Moving Average (STARMA) models. Many discrete-time, discrete-space spatio-temporal processes can be analyzed by using STARMA theorems.

The integrated statistical computing environment, IEAST (Integrated Environment for Analyzing STARMA models), was designed for analyzing and modeling the stochastic processes based on general Space-Time AutoRegressive Moving Average (STARMA) models in two-dimensional gridded space. Current version only support spatial dimension N $\times$ N, that is, the numbers of cells in X and Y direction are equal. IEAST takes data files (**.dat**), spatial weight files (.wet), program files (.pgm), and spatial/temporal configurations (in setup menu) as inputs. It outputs in plots, text numerical results, and equations (see Figure 1.1).

This system was developed on RedHat Linux 9.0 using the GNU OCTAVE v2.1.40, which is a high-level language primarily intended for numerical matrix computations. OCTAVE provides a convenient command line interface for numerically solving linear and nonlinear problems, and for performing other numerical operations using a language that is largely compatible with MatLab®. Because of source code compatibility, IEAST can be easily run under Windows/Unix/MacOS without modifications.

To provide flexibility, there are two user interfaces provided in IEAST: **menu-driven mode** and **interpreter mode**(programming mode). In menu-driven mode, users can do the modeling or analyzing procedure by selecting a hierarchical menu commands. Users need to control the flow of the procedure by themselves. In programming mode, there is a simple STARMA programming language provided. Highly integrated instructions are provided for users to compose STARMA modeling/analyzing procedures. Users can design an efficient and autonomous modeling procedure for specific applications by simply combining instructions and flow controls. It is common to

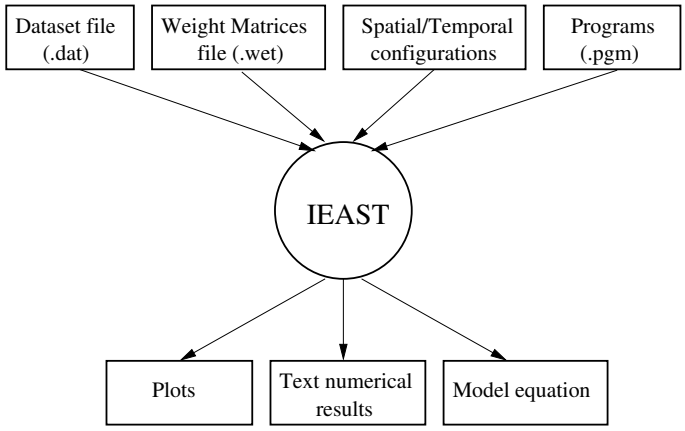

Figure 1.1: Inputs and outputs of IEAST.

implement a STARMA modeling procedure in 20 lines of codes. This programmability is especially useful when iterative procedure is necessary during analysis.

This system is designed for two-dimensional STARMA analysis. However, by carefully designing the spatial weight matrices, this system can be adapted to one-dimensional or even three-dimensional systems.

If the computer memory and computing power are sufficient, there is no limitations to the spatial dimensions of the dataset explored. But, if the spatial dimension is too small, the marginal effect at the boundaries would be significant and should be taken into account.

In univariate ARMA modeling, a well-known modeling procedure is the *three-stage iterative modeling*, which is referred to as the Box-Jenkins method. The space-time extension of the three-stage iterative modeling procedure was first presented in 1980 by Pfeifer and Deutsch. The modeling procedure used in IEAST is a refinement of that of Pfeifer and Deutsch (see Figure 2.4).

This document corresponds to IEAST version 1.30.01.

1.2 Quick Start - A Simple Example

In this section, a simple example is used to demonstrate how to use IEAST to analyze and model a dataset. The dataset `demo-1.dat` used for this demonstration was generated by simulation with the following STARMA model and parameters.

$$\begin{aligned}\mathbf{Z}(t) &= 0.30\mathbf{Z}(t-1) + 0.20\mathbf{W}^{(1)}\mathbf{Z}(t-1) + 0.10\mathbf{W}^{(2)}\mathbf{Z}(t-1) \\ &\quad -0.20\mathbf{Z}(t-2) - 0.10\mathbf{W}^{(1)}\mathbf{Z}(t-2) - 0.07\mathbf{W}^{(2)}\mathbf{Z}(t-2) + \boldsymbol{\varepsilon}(t)\end{aligned}$$

where

$$E[\boldsymbol{\varepsilon}(t)] = \mathbf{0}$$

$$Var[\boldsymbol{\varepsilon}(t)] = 0.00125\mathbf{I}.$$

The following are the steps needed for STARMA analysis and modeling of the given dataset.

1. **Invoking IEAST**: On most systems, the way to invoke IEAST is with the command '`octave ieast.m`'. Under UNIX system, you can simply type '`ieast.m`' under Octave prompt. IEAST displays a short message and then the main menu. If you get into trouble, you usually can interrupt IEAST by type `Control-C`. Doing this will normally return you to your operating system. To exit IEAST, select `[ Exit ]` under the main menu, or press several `<Enter>` till exit.

   Now, run GNU Octave under your OS (Unix/Windows/Mac), then enter the main program `ieast` (be sure you are in the same directory as IEAST system).

```
$ octave

GNU Octave, version 2.1.36 (i686-pc-cygwin).
Copyright (C) 1996, 1997, 1998, 1999, 2000, 2001, 2002 John W. Eaton.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTIBILITY or
FITNESS FOR A PARTICULAR PURPOSE.  For details, type 'warranty'.

Report bugs to <bug-octave@bevo.che.wisc.edu>.

>> dir
ieast-v1.30.01
>> cd ieast-v1.30.01
>> ieast

[IEAST v1.30.01 - STARMA Modeling & Analysis]

================= Main Menu ================
  [ 1] Setup
  [ 2] Data Preprocessing
  [ 3] AutoCorrelation Analysis
  [ 4] Model Identification
  [ 5] Parameter Estimation
  [ 6] Diagnostic Analysis
  [ 7] Forecasting
  [ 8] Preference
  [ 9] Interpreter
  [10] Exit
============================================

Selection: (or <Enter> to exit) 1
```

2. **Load dataset**: Under menu `[1] Setup`, load the dataset for analysis by selecting `[ Load IEAST Dataset Files ]`.

```
============== [ Setup ] ==============
 [ 1] > Space-time dataset
 [ 2] > Spatial correlation structure
 [ 3] > Information of datasets
 [ 4] > Return
=======================================

Selection: (or <Enter> to exit) 1

================== [ Space-time dataset ] ==================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
```

```
 [ 6] >> Return
============================================================

Selection: (or <Enter> to exit) 1

Dataset files in current directory...
-------------------------------------
(1)  demo-1.dat
(2)  NA_Precipitation.dat
(3)  NOAA_Temperature.dat
(4)  US_Precipitation.month.dat
-------------------------------------

Which Dataset file to load? (Select a number): (1..4) 1

"demo-1.dat" is being loaded ...done.

[ File name:                              demo-1.dat ]
[ Data size:                  Space=10x10, Time=100   ]
[ Maximum:                    0.139821                ]
[ Minimum:                    -0.131962               ]
[ Sample Mean:                0.000511                ]
[ Sample Variance:            0.037541                ]
[ Value for data missing:     -99.900000              ]
[ # of Missing Data:          0                       ]
```

3. **Load spatial weighting structure**: Select `[ Load Spatial Weighting From Files ]` and choose a file from current directory.

```
============== [ Setup ] ==============
 [ 1] > Space-time dataset
 [ 2] > Spatial correlation structure
 [ 3] > Information of datasets
 [ 4] > Return
=======================================

Selection: (or <Enter> to exit) 2

==== [ Spatial correlation structure ] ====
 [ 1] >> Load from file (.wet)
 [ 2] >> Define by users
 [ 3] >> Return
============================================

Selection: (or <Enter> to exit) 1

Spatial correlation structure files in current directory...
-----------------------------------------------------------
(1)  UF-S10-10x10NBPP.wet
(2)  UF-S10-16x16NBPP.wet
(3)  UF-S10-20x20NBPP.wet
(4)  UF-S6-14x14NBPP.wet
-----------------------------------------------------------

Which Spatial correlation structure file to load? (Select a number): (1..4) 1

Loading spatial definition file "UF-S10-10x10NBPP.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]
```

```
 0  0  0 10  9  9  9 10  0  0  0
 0 10  9  8  7  7  7  8  9 10  0
 0  9  8  6  5  5  5  6  8  9  0
10  8  6  5  4  3  4  5  6  8 10
 9  7  5  4  2  1  2  4  5  7  9
 9  7  5  3  1  0  1  3  5  7  9
 9  7  5  4  2  1  2  4  5  7  9
10  8  6  5  4  3  4  5  6  8 10
 0  9  8  6  5  5  5  6  8  9  0
 0 10  9  8  7  7  7  8  9 10  0
 0  0  0 10  9  9  9 10  0  0  0

[ Spatial Relation Matrix ] (SOD_MAX_S=10)

.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.050 .083 .063 .250 .250 .000 .250 .250 .063 .083 .050
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000
```

4. **Space-time autocorrelation analyses**: Go back to main menu (by pressing `<Enter>` in menu `[1] Setup`), select `[3] AutoCorrelation Analysis`, and then select `[ AutoCorrelation (ACF) ]` and wait results for few minutes(if your computer faster enough). Do the similar steps to calculate PACF by selecting `[ Partial AutoCorrelation (PACF) ]`.

```
[IEAST v1.30.01 - STARMA Modeling & Analysis]
=============== [ Main Menu ] ===============
 [ 1] Setup
 [ 2] Data Preprocessing
 [ 3] Correlation Analyses
 [ 4] Model Identification
 [ 5] Parameter Estimation
 [ 6] Diagnostic Analysis
 [ 7] Forecasting
 [ 8] Preference
 [ 9] Interpreter
 [10] Exit
=============================================

Selection: (or <Enter> to exit) 3

========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
=============================================

Selection: (or <Enter> to exit) 1

? Remove mean of the variable Z (=0.0005106068) for autocorrelation? (Y/N) [Y]
### Mean has been removed for calculating autocorrelations.
```

```
? Max. spatial lag: [4] (0..10)
### MAX_S_LAG(=4) unchanged.

? Max. temporal lag: [20] (1..99)
### MAX_T_LAG(=20) unchanged.

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Autocorrelation of Z, Rho(s, l, k=0)
-----------------------------------
 S-lag  0     1     2     3     4
T-lag-------------------------------
  1   .259  .112  .078  .004  .002
  2  -.125  .012  .003  .014  .005
  3  -.100 -.048 -.019  .005 -.021
  4  -.020 -.013 -.008 -.006 -.025
  5   .029  .019 -.006  .001 -.016
  6   .029  .013  .010 -.004 -.008
  7   .020  .014  .015  .006  .006
  8   .016  .018  .020  .020  .010
  9   .006  .003  .010  .007  .026
 10  -.004 -.018  .007 -.003  .001
 11  -.023 -.013 -.004  .003 -.023
 12  -.020 -.001 -.008  .001 -.022
 13   .008  .009 -.019  .005 -.014
 14   .006 -.004 -.001 -.019  .012
 15   .000  .011  .027 -.029  .029
 16   .006  .015  .000 -.011  .009
 17  -.002  .015 -.010 -.001 -.006
 18  -.006  .010 -.006 -.014  .002
 19  -.012  .006  .001 -.015  .002
 20  -.002 -.003  .019 -.018  .001

Press any key to continue...

========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 2

? Remove mean of the variable Z (=0.0000000000) for autocorrelation? (Y/N) [N]

? Max. spatial lag: [4] (0..10)
### MAX_S_LAG(=4) unchanged.

? Max. temporal lag: [20] (1..99)
### MAX_T_LAG(=20) unchanged.
```

```
Calculating STPACF...
|--------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Partial Autocorrelation of Z, Phi(s,l)
---------------------------------
 S-lag  0     1     2     3     4
T-lag-----------------------------
  1   .259  .161  .093 -.020 -.010
  2  -.206 -.073 -.063  .028 -.003
  3  -.009 -.006  .033 -.001 -.053
  4  -.014  .031  .009  .001 -.017
  5   .020  .015 -.018  .013 -.025
  6   .009 -.008  .015 -.013 -.017
  7   .017  .026  .011  .015  .013
  8   .015  .015  .021  .024 -.003
  9   .006 -.013 -.003 -.012  .051
 10  -.001 -.025  .020  .002 -.027
 11  -.020 -.002  .002  .015 -.024
 12  -.010  .002  .002 -.004 -.025
 13   .011  .016 -.030  .011 -.019
 14  -.008 -.027  .006 -.044  .032
 15   .001  .041  .042 -.031  .040
 16   .008  .004 -.038 -.020 -.012
 17  -.005  .029  .002 -.008  .005
 18  -.002  .018 -.003 -.042  .019
 19  -.010  .018  .003 -.016 -.008
 20   .003 -.005  .031 -.034  .011
```

5. **Plotting ACF/PACF**: The results of ACF and PACF can also be plotted versus temporal/spatial lags as shown in Figure 1.2 by selecting `[ Plot Correlations versus T-Lag/S-Lag ]`.

```
========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 6

  Which correlation to plot:
    [1] STACF  of Z
    [2] STPACF of Z
    [3] STACF  of Y
    [4] STPACF of Y
    [5] STXCF  of Z and Y
    [6] STPXCF of Z and Y
    [7] Extended STXCF
  Selection: (1..7) 1

? Plotting from spatial lag: [0] (0..4)
? Plotting up to spatial lag: [4] (0..4)
? Plotting from temporal lag: [1] (1..20)
? Plotting up to temporal lag: [20] (1..20)

  Plotting versus:
    [1] Temporal lags
```

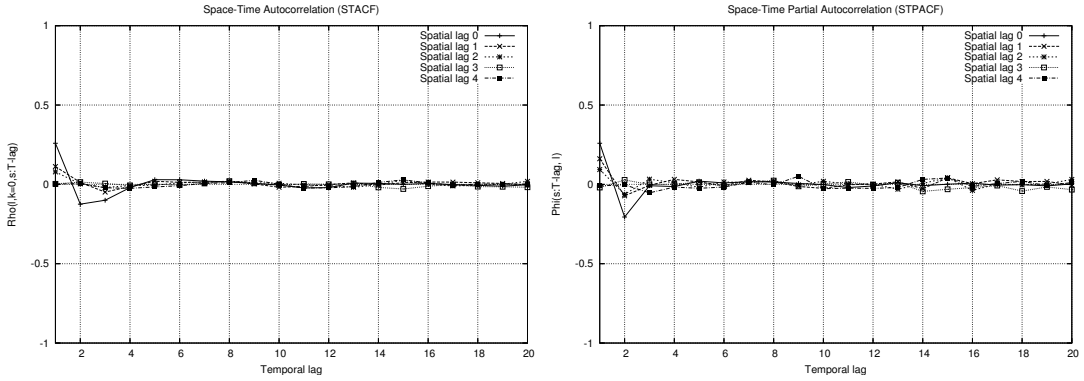

Figure 1.2: Space-time ACF and PACF for STAR(2,2)

```
    [2] Spatial lags
  Selection: (1..2) 1

### Plotting ACF of Z versus T-Lag...
### The plot is also saved as an eps file (ACF_PLOT_T.eps).

========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 6

  Which correlation to plot:
    [1] STACF  of Z
    [2] STPACF of Z
    [3] STACF  of Y
    [4] STPACF of Y
    [5] STXCF  of Z and Y
    [6] STPXCF of Z and Y
    [7] Extended STXCF
  Selection: (1..7) 2

? Plotting from spatial lag: [0] (0..4)
? Plotting up to spatial lag: [4] (0..4)
? Plotting from temporal lag: [1] (1..20)
? Plotting up to temporal lag: [20] (1..20)

  Plotting versus:
    [1] Temporal lags
    [2] Spatial lags
  Selection: (1..2) 1

### Plotting PACF of Z versus T-Lag...
### The plot is also saved as an eps file (PACF_PLOT_T.eps).
```

6. **Select model type/orders**: According to the autocorrelation results, we can decide the

tentative model type and orders either automatically or artificially. In this case, the dataset was given automatically the model STAR(S=2,T=2) by IEAST (by choosing `[4] Model Identification` and then `[ Automatic Identification ]`).

```
[IEAST v1.30.01 - STARMA Modeling & Analysis]
=============== [ Main Menu ] ===============
 [ 1] Setup
 [ 2] Data Preprocessing
 [ 3] Correlation Analyses
 [ 4] Model Identification
 [ 5] Parameter Estimation
 [ 6] Diagnostic Analysis
 [ 7] Forecasting
 [ 8] Preference
 [ 9] Interpreter
 [10] Exit
=============================================

Selection: (or <Enter> to exit) 4

=========== [ Model Identification ] ===========
 [ 1] > Automatic Identification (Type,Orders)
 [ 2] > Artificial Identification (Type,Orders)
 [ 3] > Parameter Masking
 [ 4] > ------
 [ 5] > Return
================================================

Selection: (or <Enter> to exit) 1

### Model type is suggested to be STAR with
### Max. Temporal Order = 2, Max. Spatial Order = 2.
```

7. **Estimate model parameters**: Under the menu `[5] Parameter Estimation`, the model parameters can be estimated as shown below.

```
[IEAST v1.30.01 - STARMA Modeling & Analysis]
=============== [ Main Menu ] ===============
 [ 1] Setup
 [ 2] Data Preprocessing
 [ 3] Correlation Analyses
 [ 4] Model Identification
 [ 5] Parameter Estimation
 [ 6] Diagnostic Analysis
 [ 7] Forecasting
 [ 8] Preference
 [ 9] Interpreter
 [10] Exit
=============================================

Selection: (or <Enter> to exit) 5

=================== [ Parameter Estimation ] ===================
 [ 1] > Pre-estimate Model Param   -- Linear (STAR)
 [ 2] > Pre-estimate Model Param   -- Non-linear (STMA,STARMA)
 [ 3] > Pre-estimate Model Param   -- From STACF/STPACF
 [ 4] > Pre-estimate Model Param   -- Specified by users
 [ 5] > Estimate Model Param       -- Fixed SRM
 [ 6] > Estimate SRM               -- Fixed Model Param
 [ 7] > Estimate SRM & Model Param -- Alternatively
 [ 8] > Return
```

```
================================================================

Selection: (or <Enter> to exit) 1

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|--------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|
Phi_Now =

   0.31   0.22   0.13
  -0.22  -0.08  -0.06

Press any key to continue...

================== [ Parameter Estimation ] ==================
 [ 1] > Pre-estimate Model Param   -- Linear (STAR)
 [ 2] > Pre-estimate Model Param   -- Non-linear (STMA,STARMA)
 [ 3] > Pre-estimate Model Param   -- From STACF/STPACF
 [ 4] > Pre-estimate Model Param   -- Specified by users
 [ 5] > Estimate Model Param       -- Fixed SRM
 [ 6] > Estimate SRM               -- Fixed Model Param
 [ 7] > Estimate SRM & Model Param -- Alternatively
 [ 8] > Return
================================================================

Selection: (or <Enter> to exit) 5

[ Spatial Relation Matrix ] (SOD_MAX_S=10)

.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.050 .083 .063 .250 .250 .000 .250 .250 .063 .083 .050
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000

? Spatial dimension: NxN, N = [10] (4..100)

------- Iter#1:(Fri Aug 06 13:01:49 2004) -------

ePHI =

   0.31   0.22   0.13
  -0.22  -0.08  -0.06

Advancement of Average Parameter SSE = 0.000001
Time elapsed = 0:1:9.91

Z(t) =
( 0.31)*Z(t-1) + ( 0.22)*W1*Z(t-1) + ( 0.13)*W2*Z(t-1) +
(-0.22)*Z(t-2) + (-0.08)*W1*Z(t-2) + (-0.06)*W2*Z(t-2) +
e(t)
```

The model equation in the result from STARMA modeling is pretty close to the one used to simulate. It is a good result, actually. This is only a simple example for demonstration. The later chapters will introduce these procedures in more details.

1.3 Features

IEAST version 1.30.01is mainly for STARMA modeling and analysis, and features:

- A true spatial and temporal analysis software.
- Full configurability: The spatial and temporal correlation structures can be specifically designed by users.
- Interactive programming environment: A script interpreter is provided. Highly integrated instructions are provided for simplifying programming efforts and improving efficiency of analysis and modeling.
- Estimation of spatial weighting structure: The spatial weighting structure can be retrieved and estimated from given datasets.
- Flexibility of defining spatial structure: A three-step definition for spatial structure makes IEAST more flexible in the definition of spatial structure, i.e. $SOD \rightarrow SRM \rightarrow SWM$.
- Improved estimation algorithm: The introduction of space-time extension of Hannan-Rissanen algorithm can greatly reduce the estimation time and possibility of converging to wrong results.
- Improved diagnostic measures: The space-time AIC/BIC provide a good criteria for the selection of model type and orders, especially useful in programming environment. The statistical significance can also be evaluated for each parameter.
- Cross correlation analysis provided for analyses of two space-time variables.
- 2D/3D plotting abilities.

1.4 Requirements

- Software
  - IEAST version 1.30.01 or later.
  - GNU Octave v2.0.16 or later. (download at `http://www.octave.org/`)
  - GNUPLOT v3.8 or later (see `http://www.gnuplot.info/`).
- Hardware: IEAST has very low hardware dependency. It can be ported to any kind of platforms which can run GNU Octave. However, this is a computing-intensive software. For most datasets with spatial dimension higher than 20x20, the following specification is suggested.
  - Processor: AMD K7 1GHz equivalent or higher.
  - Memory: 128MB or bigger.
  - Disk Space: 100MB.

1.5 Installation

The system is highly suggested to be installed and run under a unix operating system (or variants), in which IEAST is well-tested. If you do not have unix, you can either install a Linux operating system (free download at `http://www.mandrake.org/`) or install the Linux Win32 simulator CYGWin (it simulates a Linux environment under all Windows systems, free install can be found at `http://www.cygwin.org/`). The instructions for install Octave on Windows can be found at `http://octave.sourceforge.net/Octave_Windows.htm`). For unix installations, follow the instructions below.

1. Install GNU Octave into your unix system.
2. Install GNUPLOT into your unix system.
3. Give commands under prompt as shown below.

```
$ mkdir <Your_Working_Directory>
$ cd <Your_Working_Directory>
$ gunzip ieast-v1.xx.xx.tar.gz
$ tar xvf ieast-v1.xx.xx.tar
```

4. To start, give the command 'octave ieast.m'

1.6 Performance

The modeling performance of IEAST using various model type selection criteria in 150 Monte Carlo simulated datasets of STAR, STMA and mixed processes (50 datasets for each) is shown in Table 1.1. We used four different criteria (variance of residuals, AICC only, BIC only, and -AICC*BIC), respectively. The parameters for generating datasets were randomly chosen, with higher weights on the parameters having lower spatial or temporal orders.

As expected, using AICC or variance has a bias toward overparameterized mixed models. BIC has much better performance than AICC. All 50 simulated STAR datasets were correctly classified as STAR. The accuracies of type determination are 86% and 66% for datasets based on STMA and mixed models, respectively. Using the product of AICC and BIC as the criterion can give even better performance, where the accuracies can reach at least 78% for all types. These criteria provide reliable quantitative measures for determining model choice, whereas previous researches relied on subjective qualitative behaviors of STACF and STPACF.

1.7 Menu System

Current version (version 1.30.01) of IEAST is designed as a menu-driven selection system (if not in the interpreter). The functionalities for analysis are distributed in this menu system. Users need to step-by-step setup and analyze dataset through the menus following a modeling procedure. The first two levels of the menu system is shown in Figure 1.3. In all menu selection, if nothing entered but the key `<Enter>`, the default values (quoted in '[ ]', e.g. [3]) will be used or actions canceled or return to previous level of the menu.

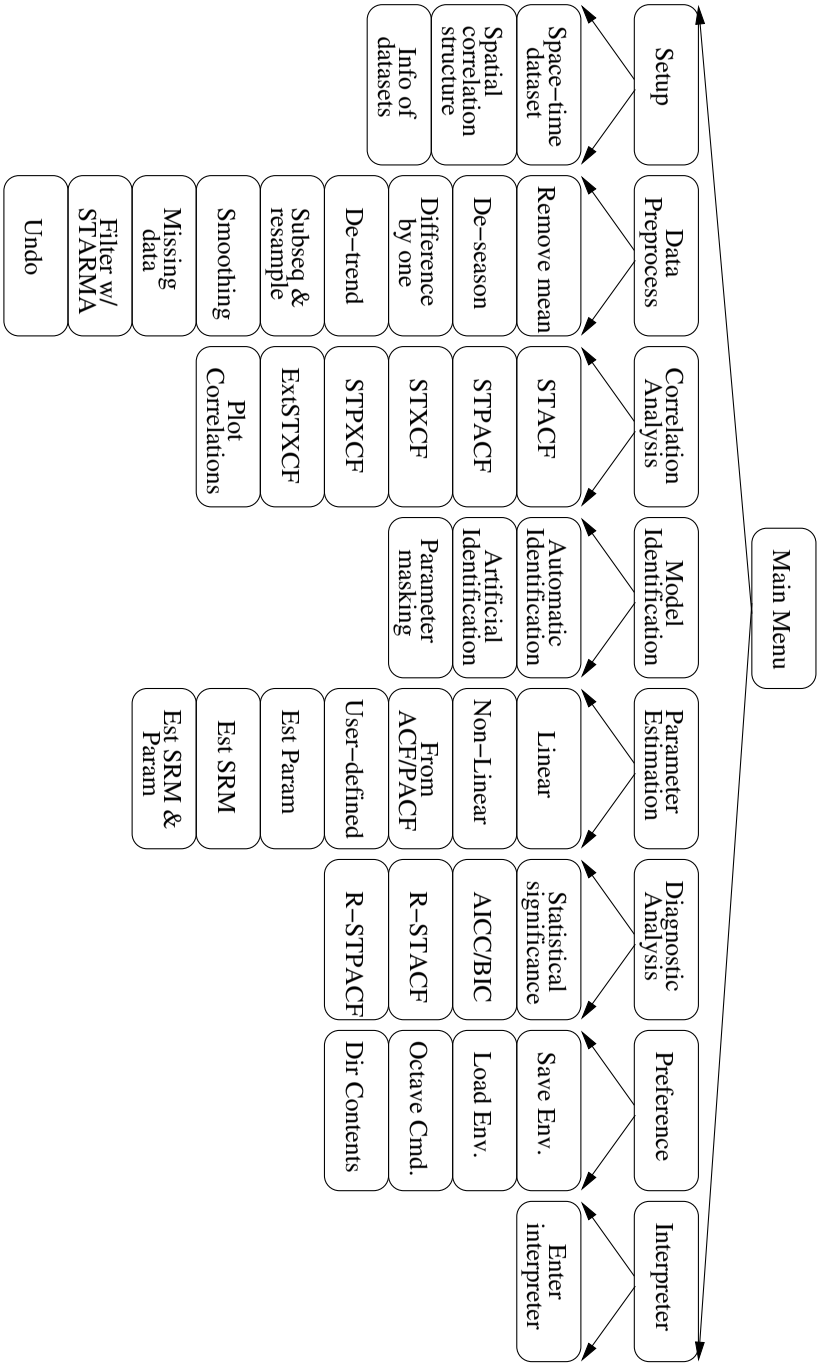

Figure 1.3: The first two levels of the menu system in IEAST

Table 1.1: Accuracies of model type selection using variances, AICC, BIC, and -AICC*BIC based on 150 Monte Carlo simulated datasets.

Using variances

Simulated datasets based on/	Datasets identified as STAR	STMA	Mixed
STAR	4%	0%	96%
STMA	4%	6%	90%
Mixed	8%	2%	90%

Using AICC

Simulated datasets based on/	Datasets identified as STAR	STMA	Mixed
STAR	16%	0%	84%
STMA	4%	6%	90%
Mixed	8%	2%	90%

Using BIC

Simulated datasets based on/	Datasets identified as STAR	STMA	Mixed
STAR	100%	0%	0%
STMA	4%	86%	10%
Mixed	18%	16%	66%

*Using -AICC*BIC*

Simulated datasets based on/	Datasets identified as STAR	STMA	Mixed
STAR	100%	0%	0%
STMA	4%	78%	18%
Mixed	16%	4%	80%

1.8 Conventions

The following should be kept in mind.

1. Because of being developed on GNU Octave, the IEAST's formats of many outputs/inputs from/to IEAST are the same as that of Octave.
2. Currently, the only user interface supported is text mode menu-driven.
3. The spatial dimension has to be N by N, (e.g. 10x10 array of cells), that is the arrangement of cells for analysis have to be square. The support for N by M is still under construction.
4. IEAST system, your data files, and weight matrix files are all located in the same working directory.

Chapter 2

STARMA Models and Modeling

2.1 Statistical Modeling

Statistical modeling is an important procedure for analyzing stochastic systems because it allows us to connect the dataset in reality to theoretical processes. Furthermore, these theoretical models can be used to forecast or to control the future behavior of a realistic ecosystem. Figure 2.1 shows how a model is used to represent the data-generating process of interest. Data and knowledge of the underlying process are used to identify a statistical model. While modeling, a dataset is used to fit this hypothetical model right after the model type and orders are determined. The parameters of the model are estimated during the fitting process. Then, adequacy of this statistical model can be evaluated. Finally, this statistical model can be used for forecasting, analysis, control or other purposes. The objective is to find a "parsimonious" model with the smallest number of parameters needed to fit adequately the patterns in the dataset.

Box and Jenkins [3] examined the use of the ARMA model for analyzing univariate time series data and generating short-term forecasts. Pfeifer and Deutsch extended the Box-Jenkins' modeling procedure to the space-time ARMA case, i.e. STARMA models [10, 12, 11, 13].

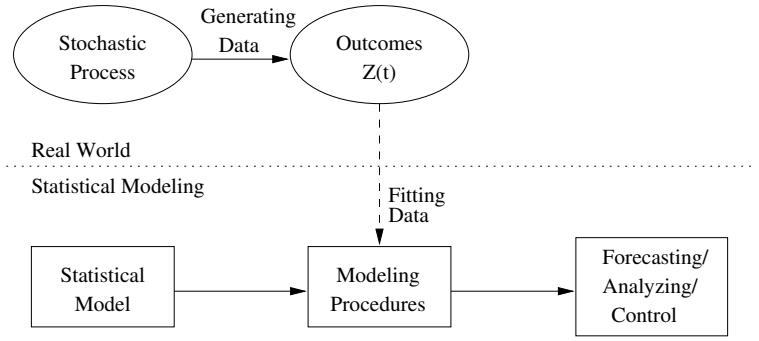

Figure 2.1: Relationship between data-generating and statistical modeling procedures

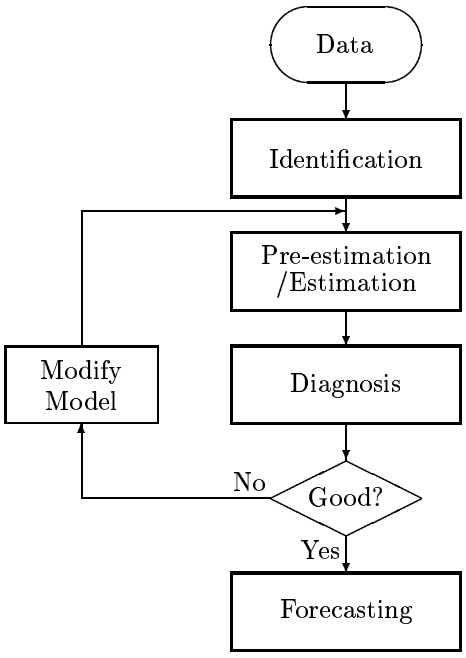

Figure 2.2: Box-Jenkins Modeling

2.2 Box-Jenkins Modeling

The Box-Jenkins' modeling procedure involves identifying an appropriate ARMA model, fitting it to the dataset, and using the fitted model for forecasting. One of the attractive features of the Box-Jenkins method to forecasting is that ARMA processes are a very rich class of possible models and it is usually possible to find a model which provides an adequate description for the data.

The Box-Jenkins' modeling procedure involves iterative steps of *model identification*, *parameter estimation*, *model diagnosis* and *forecasting*. Figure 2.2 outlines a modified Box-Jenkins' procedure for searching an adequate STARMA model. Before estimation, a step *pre-estimation* is suggested to be done in advance. Each of these steps are discussed and illustrated in the sections that follow.

2.3 STARMA Model Definition

The STARMA model class is characterized by linear dependence lagged both in space and time. Here, we follow primarily the notation of Pfeifer's [10].

STARMA expresses $Z_i(t)$, the observation of the random variable $\mathbf{Z}$ at site i *(i=1,2,...,N)* and time t, as a weighted linear combination of past observations and random noise inputs, which may be lagged both in space and time.

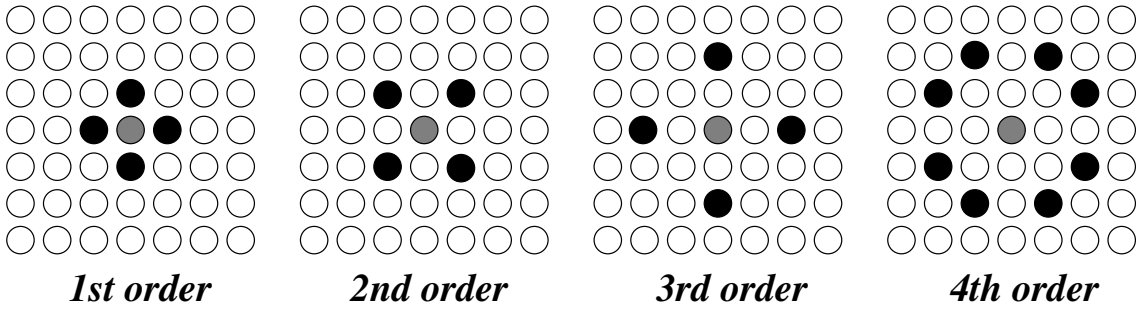

Figure 2.3: An example of definition of spatial orders in 2D systems

Let $\boldsymbol{Z}(t)$ be the $N \times 1$ vector of observed values at time t, i.e.

$$\mathbf{Z}(t) = [Z_1(t), Z_2(t), \cdots, Z_N(t)]'$$

where N is the total number of sites in two-dimensional space.

To formulate the spatial relations among sites, weight matrices $\boldsymbol{W}^{(l)}$ are introduced as in [10]. Assume weight matrix $\boldsymbol{W}^{(l)}$ has elements $w_{ij}^{(l)}$ that are the weighting contributions of site j to site i, and which are nonzero if an only if site i and j are *l-th* order neighbors in space. Then, the general STARMA model can be expressed in the following form:

$$\mathbf{Z}(t) = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_k} \phi_{kl} \mathbf{W}^{(l)} \mathbf{Z}(t-k) - \sum_{k=1}^{q} \sum_{l=0}^{m_k} \theta_{kl} \mathbf{W}^{(l)} \boldsymbol{\varepsilon}(t-k) + \boldsymbol{\varepsilon}(t) \tag{2.1}$$

where p is the maximum autoregressive (temporal) order, q is the maximum moving average (temporal) order, λ_k is the spatial order of the k^{th} autoregressive term, m_k is the spatial order of the k^{th} moving average term, ϕ_{kl} is the autoregressive parameter at temporal lag k and spatial lag l, θ_{kl} is the moving average parameter at temporal lag k and spatial lag l, $\mathbf{W}^{(l)}$ is the $N \times N$ matrix of weights for spatial order l, and $\boldsymbol{\varepsilon}(t) = [\varepsilon_1(t), \varepsilon_2(t), \cdots, \varepsilon_N(t)]'$ is the random noise vector at time t.

The weights $w_{ij}^{(l)}$ should reflect an ordering of spatial neighbors. Figure 2.3 shows an example of a spatial order definition.

The first order neighbors (corresponding to the 1^{st} order weight matrix) are those that are closest to a given site (i.e. the small black dot in the center of Figure 2.3). The 2^{nd} order neighbors are farther away from than the 1^{st} order neighbors, but closer than the 3^{rd} order neighbors.

The model is specified as a STARMA(p, q, λ_k, m_k). That is, autoregressive terms have the maximum lag p in time and λ_k in space, and moving average terms have the maximum lag q in time and m_k in space.

There are two important subclasses of STARMA models. A stochastic process $\mathbf{Z}(t)$ is said to be a Space-Time AutoRegressive process of temporal order p and spatial order λ_k ($STAR(p, \lambda_k)$) if $q = 0$, and a Space-Time Moving Average process of temporal order q and spatial order m_k ($STMA(q, m_k)$) if $p = 0$.

The STAR process is described as

$$\mathbf{Z}(t) = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_k} \phi_{kl} \mathbf{W}^{(l)} \mathbf{Z}(t-k) + \boldsymbol{\varepsilon}(t) \tag{2.2}$$

and the STMA process is described as

$$\mathbf{Z}(t) = \varepsilon(t) - \sum_{k=1}^{q} \sum_{l=0}^{m_k} \theta_{kl} \mathbf{W}^{(l)} \varepsilon(t-k). \tag{2.3}$$

These two subclasses are essential. Not only can many practical stochastic processes be simply attributed to STAR or STMA, but also there exist primary statistical differences between them. We discuss these issues in detail later.

2.4 Elementary Statistical Tools for Analyzing STARMA Models

The following are the definitions of the elementary tools for analyzing STARMA models. These tools provide a systematic way for revealing the types and orders of the underlying model.

By the definition in [10], the space-time covariance function between l^{th} and k^{th} order neighbors at time lag s can be expressed as

$$\gamma_{lk}(s) = E_t \left[\frac{[\mathbf{W}^{(l)}\mathbf{Z}(t)]'[\mathbf{W}^{(k)}\mathbf{Z}(t+s)]}{N} \right]. \tag{2.4}$$

Based on the definition of space-time covariance $\gamma_{lk}(s)$(as in Eq. 2.4), the following two essential functions can be defined. Space-Time AutoCorrelation Function (STACF) and Space-Time Partial AutoCorrelation Function (STPACF) are elementary in determining the preliminary model type and orders from the observed dataset. Their calculations are described in [9], [5], and [12].

The STACF between l^{th} and k^{th} order neighbors at time lag s can be expressed as

$$\rho_{lk}(s) = \frac{\gamma_{lk}(s)}{\sqrt{\gamma_{ll}(0)\gamma_{kk}(0)}}. \tag{2.5}$$

Similar to the univariate ARMA case, the STPACF can be found as follows. Given s and h, which are respectively the maximum temporal and spatial orders, the BLP (Best Linear Predictor) of $\mathbf{Z}(t)$ is

$$\hat{\mathbf{Z}}(t)|_{s,h} = \sum_{k=1}^{s} \sum_{l=0}^{h} \phi_{kl} \mathbf{W}^{(l)} \mathbf{Z}(t-k). \tag{2.6}$$

For every $s = 1...p$ and $h = 0...\lambda$, a set of coefficients $\{\phi_{kl} | k = 1...s, l = 0...h\}$ can be found for some BLP $\hat{\mathbf{Z}}(t)|_{s,h}$. Thus, the STPACF is the set

$$\{\phi_{sh} | \phi_{sh} \textit{ is the last element of the set } \xi_{sh}, s = 1...p, h = 0...\lambda\} \tag{2.7}$$

where $\xi_{sh} = \{\phi_{kl} | k = 1...s, l = 0...h\}$ is the set of coefficients making $\hat{\mathbf{Z}}(t)|_{s,h}$ a BLP of $\mathbf{Z}(t)$ with the limitations of maximum temporal order s and maximum spatial order h. As shown in [9] and [12], ξ_{sh} is the solution of the space-time extension of Yule-Walker equations with maximum temporal order s and maximum spatial order h.

2.5 Assumptions

For a dataset to be meaningful to STARMA analysis, there are some assumptions to be met. First, datasets are assumed to be zero-mean and stationary. If they are not, some preprocessing or transformations such as differencing, de-seasonalizing, de-trending, or logarithmic transformation must be done before modeling. The processes to be modeled are the stochastic processes whose properties, such correlative structures, do not vary with location and time.

To be stationary and having a unique solution, just as in the univariate case, the *causality* and *invertibility* conditions of the process must be satisfied.

If $p = 0$ in Eq. 2.1, the model is a pure STMA model as shown in Eq. 2.3 and $\mathbf{Z}(t)$ is always stationary. When a STARMA model has p autoregressive terms ($p > 0$), $\mathbf{Z}(t)$ is stationary if and only if all roots of

$$det\left[\mathbf{I} - \sum_{k=1}^{p}\sum_{l=0}^{\lambda_k}\phi_{kl}\mathbf{W}^{(l)}x^{-k}\right] = 0$$

are inside the unit circle, that is, $|x| < 1$. If $q = 0$ in Eq. 2.1, the model is a pure STAR model as shown in Eq. 2.3 and $\mathbf{Z}(t)$ is always invertible. When a STARMA model has q moving average terms ($q > 0$), $\mathbf{Z}(t)$ is invertible if and only if all roots of

$$det\left[\mathbf{I} - \sum_{k=1}^{q}\sum_{l=0}^{m_k}\theta_{kl}\mathbf{W}^{(l)}x^{-k}\right] = 0$$

are inside the unit circle, that is, $|x| < 1$ [12].

It is assumed that each observation of the space-time series has the same expectation function, standard deviation, and probability distribution function. We assumed a noise error component, which is a sequence of uncorrelated random vectors with a constant distribution (Gaussian), constant variance, and zero mean.

The random noise vector $\varepsilon(t)$ is normally distributed at time t and satisfies the following characteristics:

$$\varepsilon(t) \sim N(\mathbf{0}, \sigma^2\mathbf{I}_N)$$

$$E[\varepsilon(t)\varepsilon(t+s)'] = \begin{cases} \sigma^2\mathbf{I}_N, & s = 0 \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$E[\mathbf{Z}(t)\varepsilon(t+s)'] = \mathbf{0} \quad \text{if } s > 0.$$

2.6 General STARMA Modeling Procedure for using IEAST

Based on the statistical tools implemented in IEAST, an extended STARMA modeling procedure (from Box-Jenkins modeling) is suggested while using IEAST. This is a general model procedure for all possible situations. Users do not have to follow all steps, but only need to extract the portion needed. For example, if the model type is known (from some prior knowledge of the dataset), the area enclosed by dotted line is not necessary for the modeling.

This modeling procedure is especially useful while programming using IEAST interpreter (see Chapter 10). IEAST interpreter provides necessary instruction set to construct the modeling flow. This programmability gives users best efficiency and flexibility.

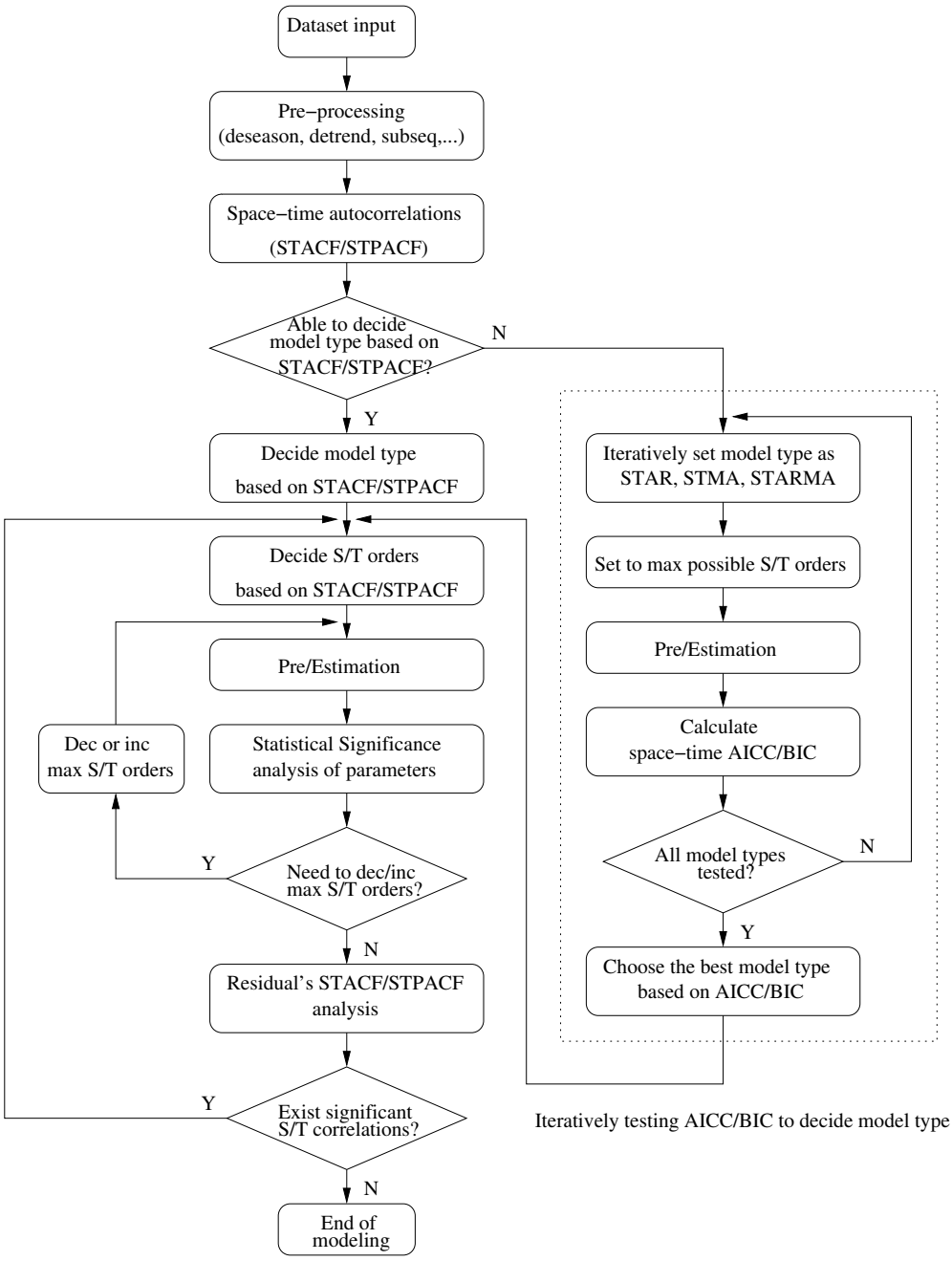

Figure 2.4: An extended STARMA modeling procedure. The area enclosed by dotted line is for the case that model type cannot be decided simply from autocorrelations. AICC/BIC of every possible model type is evaluated and compared to select the best model type.

Chapter 3

Data Preparation

The targets for STARMA analysis are space-time datasets. There are two global variables which can be loaded with space-time datasets in IEAST, i.e. the major variable (**Z**) and the secondary variable (**Y**). The major variable is involved in all space-time analyses. The secondary variable is needed only while doing *cross analyses* between two variables.

The only file format from which IEAST can directly retrieve dataset is the **octave-formated** data file **.dat**, which is a standard format for storing variables in GNU Octave. Besides, IEAST also provides some transformations from other data formats to **.dat** for further uses. There are three ways to obtain space-time datasets, which are: (1) from .dat, (2) imported from other formats, and (3) from simulation.

3.1 Load space-time dataset from .dat files

Under the menu `[1] Setup`, select `[ Space-time dataset ]`, and then `[ Load major variable (Z) from dataset file (.dat) ]`. Follow the procedure as shown below. To load secondary variable (**Y**) has a similar procedure.

```
============== [ Setup ] ==============
 [ 1] > Space-time dataset
 [ 2] > Spatial correlation structure
 [ 3] > Information of datasets
 [ 4] > Return
=======================================

Selection: (or <Enter> to exit) 1

=================== [ Space-time dataset ] ===================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
 [ 6] >> Return
=============================================================

Selection: (or <Enter> to exit) 1
```

```
Dataset files in current directory...
-------------------------------------
(1)  NA_Precipitation.dat
(2)  NOAA_Temperature.dat
(3)  US_Precipitation.month.dat
(4)  WNV_Crow_WAM_10x10full.dat
-------------------------------------

Which Dataset file to load? (Select a number): (1..4) 1

"US_Precipitation.month.dat" is being loaded ...done.

[ File name:                 US_Precipitation.month.dat ]
[ Data size:                    Space=20x20, Time=120   ]
[ Maximum:                      28.145200               ]
[ Minimum:                      0.000000                ]
[ Sample Mean:                  3.213700                ]
[ Sample Variance:              2.662913                ]
[ Value for data missing:       -99.900000              ]
[ # of Missing Data:            0                       ]
```

3.2 Import space-time datasets from other formats

IEAST supports three commonly used formats in remote sensing datasets. These files have to be in text (ASCII) and conform with one of the following formats. Importing can only work on the major variable (**Z**). Notice: The X,Y, and Time must be normalized to integers (from 1...Max). $D(X = x, Y = y, T = t)$ are the corresponding data values for location (x, y) at time t.

XY+T (.xyt) Each row is for different T. In each row, X changes first, then Y. For example, for a space NxN, the first row of your table can be: (1,1), (2,1), (3,1), ..., (N,1), (1,2), (2,2), ..., (N,2), ..., (N,N). See an example in Table 3.1.

Table 3.1: An example for format XY+T of SPACE:10x10 and TIME:1..28

$D(X=1,Y=1,T=1)$	$D(X=2,Y=1,T=1)$	.....	$D(X=10,Y=10,T=1)$
$D(X=1,Y=1,T=2)$	$D(X=2,Y=1,T=2)$	.....	$D(X=10,Y=10,T=2)$
:	:	.....	:
:	:	.....	:
$D(X=1,Y=1,T=28)$	$D(X=2,Y=1,T=28)$	.....	$D(X=10,Y=10,T=28)$

T+XY (.txy) Each column is for different T. In each column, X changes first, then Y. See an example in Table 3.2.

XYTD (.xyd) Each row is a record for some location at some time. Columns are for X, Y, Time, and Data values. The first row can be names for each field. See an example in Table 3.3.

As an example, under the menu `[1] Setup`, select `[ Space-time dataset ]`, and then `[ Import major variable (Z) from XY+T/T+XY/XYTD ]`. In the example, the imported dataset is in XYTD format, and N_SITE=100(NxN=10x10) and N_TIME=13. Because format **.dat** is

Table 3.2: An example for format T+XY of SPACE:10x10 and TIME:1..28

$D(X=1,Y=1,T=1)$	$D(X=1,Y=1,T=2)$	.....	$D(X=1,Y=1,T=28)$
$D(X=2,Y=1,T=1)$	$D(X=2,Y=1,T=2)$	.....	$D(X=2,Y=1,T=28)$
:	:	.....	:
:	:	.....	:
$D(X=10,Y=10,T=1)$	$D(X=10,Y=10,T=2)$	.....	$D(X=10,Y=10,T=28)$

Table 3.3: An example for format XYTD of SPACE:10x10 and TIME:1..28

field_X	field_Y	field_Time	field_Data
1	3	3	0.1
1	4	2	-0.5
2	1	2	1.4
:	:	:	:
:	:	:	:
9	10	28	0.2

convenient, remember to save your imported dataset to a **.dat** file for next time use by `[ Save dataset from major variable (Z) to file (.dat) ]`. Following the procedure below.

```
=================== [ Space-time dataset ] ===================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
 [ 6] >> Return
=============================================================

Selection: (or <Enter> to exit) 3

The file to be imported must be tab-separated (.txt) or comma-separated (.csv) and
in one of the formats: XY+T, T+XY, or XYTD.
  Select file format:
    [1] Tab-separated (.txt)
    [2] Comma-separated (.csv)
  Selection: (1..2) 1

Tab-separated files in current directory...
-------------------------------------------
(1)  CCF-varCrow-XYTD.txt
(2)  CCF-varHuman-XYTD.txt
(3)  DeerTB-10x10-XYTD.txt
(4)  NOAA_Temperature-XYT.txt
-------------------------------------------

Which Tab-separated file to load? (Select a number): (1..4) 1

  Select data format:
    [1] XY+T
    [2] T+XY
```

```
    [3] XYTD
  Selection: (1..3) 3

Example (XYTD): If the spatial ...

              :
              :
              :

The following is the first line in the file.

X       Y       N       T

Are they the field names? (Y/N) [Y]

(1) X
(2) Y
(3) N
(4) T

which field is for coordinate-X?(1..4) 1

which field is for coordinate-Y?(1..4) 2

which field is for Time?(1..4) 4

which field is for Data?(1..4) 3

Reading records from file...done.

Dimension of the gridded space (NxN): N=[10](10..100)

Numer of time points: T=[28](28..10000)
How to deal with repeated records at the same location and time?
  To:
    [1] discard the latter
    [2] replaced with the latter
    [3] sum up them
  Selection: (1..3) 3

There is some cell(s) in space/time not being assigned values.
  Fill them with:
    [1] zeros
    [2] Default missing marks (MISSING=-99)
    [3] -1
    [4] User-defined values
  Selection: (1..4) 1

### Empty cells are filled with 0.

[ File name:                      CCF-varCrow-XYTD.txt ]
[ Data size:                   Space=10x10, Time=28    ]
[ Maximum:                     27.000000               ]
[ Minimum:                     0.000000                ]
[ Sample Mean:                 0.692143                ]
[ Sample Variance:             1.545331                ]
[ Value for data missing:      -99.900000              ]
[ # of Missing Data:           0                       ]
```

3.3 Simulation Data

IEAST can also generate Monte Carlo simulated STARMA datasets. The following example gives a simulation for a dataset based on the facts: Space: 10x10 cells, Time: 80 periods, Max. Temporal Order = 2, Max. Spatial Order = 1, and the model equation

$$\mathbf{Z}(t) = (+0.40)\mathbf{Z}(t-1) + (-0.30)\mathbf{W}^{(1)}\mathbf{Z}(t-1) + (+0.20)\mathbf{Z}(t-2) + (-0.10)\mathbf{W}^{(1)}\mathbf{Z}(t-2) + \varepsilon(t).$$

Again, remember to save the dataset to a **.dat** file after simulation is done.

```
================== [ Space-time dataset ] ==================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
 [ 6] >> Return
============================================================

Selection: (or <Enter> to exit) 4

### Weight matrices are not yet specified. Press any key to load .wet file.

Spatial correlation structure files in current directory...
------------------------------------------------------------
(1) UF-S6-10x10NBPP.wet
(2) UF-S6-14x14NBPP.wet
(3) UF-S6-20x20NBPP.wet
(4) UF-S8-16x16NBPP.wet
------------------------------------------------------------

Which Spatial correlation structure file to load? (Select a number): (1..4) 1

Loading spatial definition file "UF-S6-10x10NBPP.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]

 0  6  5  5  5  6  0
 6  5  4  3  4  5  6
 5  4  2  1  2  4  5
 5  3  1  0  1  3  5
 5  4  2  1  2  4  5
 6  5  4  3  4  5  6
 0  6  5  5  5  6  0

[ Spatial Relation Matrix ] (SOD_MAX_S=6)

.000 .125 .063 .063 .063 .125 .000
.125 .063 .125 .250 .125 .063 .125
.063 .125 .250 .250 .250 .125 .063
.063 .250 .250 .000 .250 .250 .063
.063 .125 .250 .250 .250 .125 .063
.125 .063 .125 .250 .125 .063 .125
.000 .125 .063 .063 .063 .125 .000

? Spatial dimension: NxN, N = [10] (4..100)
? Model type: (1:STAR,2:STMA,3:STARMA) [N/A] (1..3) 1
? Max. temporal order: [N/A] (1..20) 2
```

```
? Max. spatial order: [N/A] (0..20) 1
? # of periods in time of the dataset: [N/A] (1..1000) 80

-------------- Current Configurations --------------
[ Spatial dimension (N_SPACE):          10 x 10 ]
[ Number of samples in time (N_TIME):   80      ]
[ Model type (MODEL):                   STAR    ]
[ Model Max. T order (MODEL_MAX_T):     2       ]
[ Model Max. S order (MODEL_MAX_S):     1       ]
[ Spatial Order Definition ]

 0  6  5  5  5  6  0
 6  5  4  3  4  5  6
 5  4  2  1  2  4  5
 5  3  1  0  1  3  5
 5  4  2  1  2  4  5
 6  5  4  3  4  5  6
 0  6  5  5  5  6  0

[ Spatial Relation Matrix ] (SOD_MAX_S=6)

.000 .125 .063 .063 .063 .125 .000
.125 .063 .125 .250 .125 .063 .125
.063 .125 .250 .250 .250 .125 .063
.063 .250 .250 .000 .250 .250 .063
.063 .125 .250 .250 .250 .125 .063
.125 .063 .125 .250 .125 .063 .125
.000 .125 .063 .063 .063 .125 .000

[ Spatial Weighting Matrices (SWM):     Ready   ]
-----------------------------------------------------

Are these configurations good?  (Y/N) [Y]

? Mean of random noise: [0.000000] (-100..100)
? Variance of random noise: [0.00125] (-100..100)
? Upper limit of values of Z:(-1e10~+1e10) [No Limit] (-100000000..100000000)
? Lower limit of values of Z:(-1e10~+1e10) [No Limit] (-100000000..100000000)
? How many iterations to simulate? [1000] (1..100000000)

### Please specify the parameters for this STAR model:

Z(t) =
(*****)*Z(t-1) + (*****)*W1*Z(t-1) +
(*****)*Z(t-2) + (*****)*W1*Z(t-2) +
e(t)

? Phi(T=1,S=0) = (-10..10) .4
? Phi(T=1,S=1) = (-10..10) -.3

? Phi(T=2,S=0) = (-10..10) .2
? Phi(T=2,S=1) = (-10..10) -.1


### The following model is being simulated:

Z(t) =
( 0.40)*Z(t-1) + (-0.30)*W1*Z(t-1) +
( 0.20)*Z(t-2) + (-0.10)*W1*Z(t-2) +
e(t)

Simulating...STAR model 100% ...Done.
```

3.4 Dataset from Excel files and other sources

If your space-time dataset is in an Excel file, you need follow the procedures below to transform it into IEAST **.dat** format.

1. Make sure your data in Excel is arranged in one of the three ways (XY+T, T+XY, or XYTD) as described in section 3.2.
2. In Excel, save your data as an Tab-separated file (**.txt**) or Comma-separated file (**.csv**).
3. Copy the file, 'your_file.txt', for example, to IEAST working directory.
4. Following the steps as in section 3.2 to import it into the major variable **Z**.
5. Save the dataset in **Z** to an IEAST dataset file (**.dat**).

IEAST can only accept 'text' data files, not 'binary'. So, if the data files are not in 'text', users need to transform them into ASCII data files in advance.

3.5 Important Global Variables

Information stored in the following global variables is the important characteristics of your STARMA model. They are essential for all the analyses in IEAST.

Z Matrix. This is the major variable storing the main dataset. It has *N_SITE* x *N_SITE* rows and T columns.

Y Matrix. This is the secondary variable storing the auxiliary dataset. It has *N_SITE* x *N_SITE* rows and T columns. It often is used while cross analyzing between variables **Z** and **Y**.

N_TIME Integer. Total number of points in time, i.e. $T = Ncolumns(\mathbf{Z})$.

N_SITE Integer. Total number of cells in Z in two-dimensional space, i.e. $N_SITE = \sqrt{Nrows(\mathbf{Z})}$.

MODEL String. Model type. There are three subclasses (or types) in general STARMA models, i.e. STAR, STMA and STARMA(mixed model). The model type is stored in the global variable *MODEL*. In interpreter, you should assign *MODEL* a string which can be one of "STAR", "STMA" and "ARMA".

MODEL_MAX_S Integer. This is the maximum spatial order in your model. It is different from *SOD_MAX_S*. Normally, *SOD_MAX_S* is greater or equal to *MODEL_MAX_S*.

MODEL_MAX_T Integer. This is the maximum temporal order in your model.

MAX_T_LAG Integer. This is the largest lag in time while calculating autocorrelation functions. It can be changed anytime.

MAX_S_LAG Integer. This is the largest lag in space while calculating autocorrelation functions. It can be changed anytime.

SYS_MAX_S Integer. This defines the maximum spatial order supported by IEAST. Default value: 20.

SYS_MAX_T Integer. This defines the maximum temporal order supported by IEAST. Default value: 20.

SOD_MAX_S Integer. The maximum spatial order defined in Spatial Order Definition matrix. Automatically generated by IEAST for references.

SOD Matrix. Spatial Order Definition.

SRM Matrix. Spatial Relation Matrix.

SWM Matrix. Spatial Weighting Matrices, or simply Weight Matrices.

PHI Matrix. The parameters of auto-regressive terms. It is a matrix with *MODEL_MAX_T* rows and *MODEL_MAX_S* columns.

THE Matrix. The parameters of moving average terms. It is a matrix with *MODEL_MAX_T* rows and *MODEL_MAX_S* columns.

CovarTB Matrix. This is a space-time covariance look-up table for accelerating the calculations of STACF and STPACF. It is a internal global variable.

E_VARIANCE Real number. Estimated variance of random noise.

3.6 Visualize Your Dataset

IEAST provides 2D/3D graphing abilities to explore datasets and analysis results. When dataset is ready, under the menu `[1] Setup`, select `[ Information ]`. It gives a menu as the following.

View current system configuration It gives the essential parameters and characteristics of the dataset in major variable.

3D Data Plotting The dataset is displayed in 3D. While user changes the time, the 3D data graph for a given time cross section changes accordingly. See an example in Figure 3.1.

Average over space (temporal trend) The dataset is averaged over space. It turns out to be the average temporal trend (a curve).

Average over space - frequency domain The dataset is Fourier transformed after averaged over space. That is, the temporal trend is shown in frequency domain.

Average over time (spatial trend) The dataset is averaged over time. It turns out to be the average spatial trend (a surface).

Average over time - frequency domain The dataset is two-dimensional Fourier transformed after average over time. That is, the spatial trend is shown in 2D frequency domain.

Histogram Display the histograms of the dataset in space or in time.

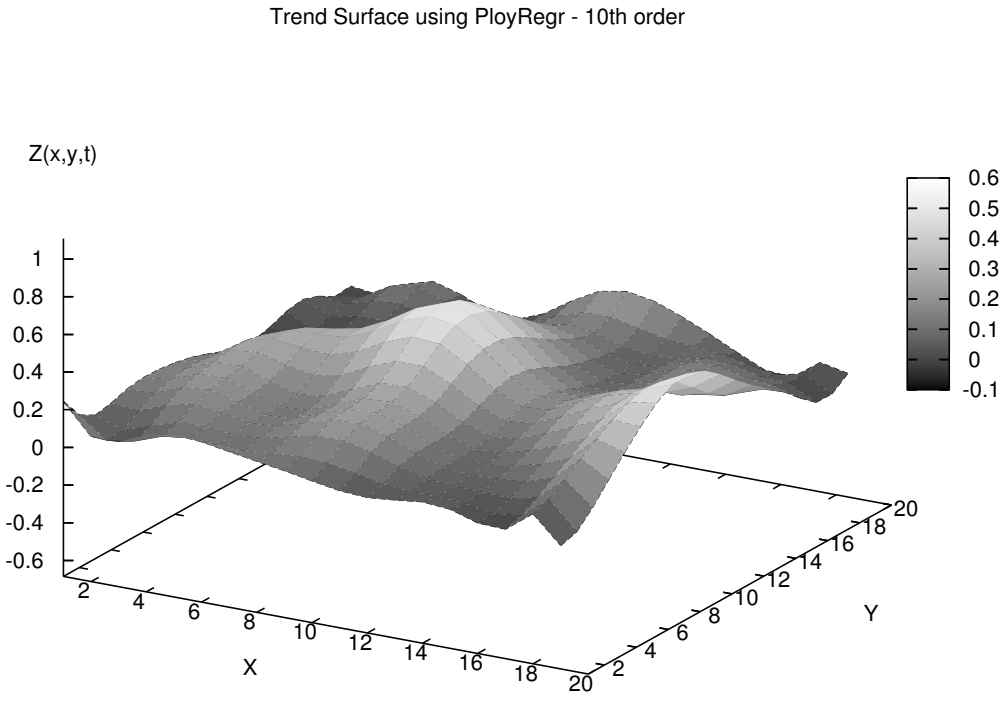

Figure 3.1: An example of 3D plotting of dataset.

Chapter 4

Spatial Correlation Structure

Beside space-time datasets, another essential step before analysis is to specify a spatial correlation structure for a dataset (in practice, it is a definition of set of weight matrices). The specification describes the relative spatial relations among cells, including the definition of spatial orders, weighting distributions, and spatial dimension. In most of practical cases, spatial correlation structures are isotropic (directionless). Thus, once the definition of spatial orders (or lags) is given, IEAST can generate an appropriate set of weight matrices for an isotropic structure.

4.1 Spatial Dimension, N_SITE and N x N

Current version (version 1.30.01) of IEAST works only on two-dimensional gridded space which have to be an array of N x N. The spatial dimension needs to be specified and stored in the system as the global variable *N_SITE*. *N_SITE* is the total number of sites(or cells) in the dataset to be analyzed, i.e. $N_SITE = N^2$.

The spatial correlation structure can be uniquely defined by a set of *weight matrices*, **SRM**. To make it clear, in IEAST, the specification of weight matrices SRM (or the specification of spatial correlation structure) consists three steps, i.e. $SOD \rightarrow SRM \rightarrow SWM$. These three matrices are stored in the same weight file `*.wet` as matrix variables **SOD**, **SRM**, and **SWM**. They together provide a sufficient information for defining a complete spatial correlation structure. These three components are described individually below.

4.2 Spatial Order Definition, SOD

It is a 21x21 matrix, and defines the spatial orders with respect to the central cell (11,11) in neighborhood. In other words, the maximum neighborhood coverage can be defined and supported by IEAST is up to 21 by 21 cells. For example, the matrix shown below gives the spatial order/lag definition (up to fifth order) of every cell. Center cell (11,11) is always zero. The cells with no relation to the center in the 21x21 matrix are 0s. *SOD* does not give any information about how important some specific cells are, but only designate a spatial order for every cell.

$$SOD = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{5} & \mathbf{4} & \mathbf{3} & \mathbf{3} & \mathbf{3} & \mathbf{4} & \mathbf{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{4} & \mathbf{3} & \mathbf{2} & \mathbf{2} & \mathbf{2} & \mathbf{3} & \mathbf{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{3} & \mathbf{2} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{2} & \mathbf{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{3} & \mathbf{2} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{2} & \mathbf{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{3} & \mathbf{2} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{2} & \mathbf{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{4} & \mathbf{3} & \mathbf{2} & \mathbf{2} & \mathbf{2} & \mathbf{3} & \mathbf{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{5} & \mathbf{4} & \mathbf{3} & \mathbf{3} & \mathbf{3} & \mathbf{4} & \mathbf{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \tag{4.1}$$

4.3 Spatial Relation Matrix, SRM

After the *SOD* defined, we need to specify the relative importance of a cell in a given spatial order. This information is stored in SRM which has the same size as that of SOD. In the last example of *SOD*, there are eight cells in first spatial order. We can give the following relative importance to the cells in first spatial order with anisotropic weights to emphasize the importance of the northern point of the central cell (weight 0.3). All weights in the same spatial order should sum to one. In IEAST, SRM can be given either by users or by system (if isotropic).

$$SRM = \begin{bmatrix}
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - & - & - & - & - & - & - & - & .1 & .3 & .1 & - & - & - & - & - & - & - & - & - \\
- & - & - & - & - & - & - & - & - & .1 & 0 & .1 & - & - & - & - & - & - & - & - & - \\
- & - & - & - & - & - & - & - & - & .1 & .1 & .1 & - & - & - & - & - & - & - & - & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & - \\
- & -
\end{bmatrix} \tag{4.2}$$

4.4 Spatial Weighting Matrices, SWM

These are the actual matrices used by IEAST for calculations. Given *SOD*, *SRM* and *N_SITE*, *SWM* can be generated by system automatically. *SWM* is normally a huge matrix with *SOD_MAX_S* (see section 3.5) rows and *N_SITE* x *N_SITE* columns.

4.5 Configure Your System

To configure your two dimensional spact-time system, under the menu `[1] Setup`, choose `[ Configure System ]` and answer the first few questions, if necessary.

? Spatial dimension: If you already have your dataset, you don't have to enter this number. It will be overwritten by the file loading thereafter. If you are going to simulate a given model, specify N here.

? Model type, Max temporal order, Max spatial order You don't have to give these three information if you are not going to simulate a dataset.

? # of samples in time of the dataset This means how many time periods in your dataset. You don't have to give this if the dataset is going to be loaded.

Thereafter, it is followed by *Spatial Structure Definition*. Here, you need to specify *SOD* and *SRM* orderly (you cannot assign values to SRM without specifying SOD in advance), and then *SWM* can be generated by IEAST. There are two options while specifying SOD.

Euclidean Distance The system semi-automatically decides the SOD based on Euclidean distance between the cell and the center. Users need to specify the value regions of the spatial order. The minimum Euclidean distance is 1. (Note:Euclidean distance is equal to $(Horizontal_Distance_to_Center)^2 + (Vertical_Distance_to_Center)^2$.)

User-defined User assigns the spatial orders in matrix SOD one by one as needed.

While defining *SRM*, there are also two options.

Uniform The system automatically decides the SRM so that all cells in each spatial order has same values and sum up to 1.0.

User-defined User assigns the values in matrix SRM one by one. Remember to keep the sum of all cells in each spatial order be 1.0.

The following is an example of specifying SOD, SRM and then generating SWM. At last, the weighting definitions are saved to a file (.wet).

```
============== [ Setup ] ==============
 [ 1] > Space-time dataset
 [ 2] > Spatial correlation structure
 [ 3] > Information of datasets
 [ 4] > Return
=======================================
```

```
Selection: (or <Enter> to exit) 2

==== [ Spatial correlation structure ] ====
 [ 1] >> Load from file (.wet)
 [ 2] >> Define by users
 [ 3] >> Return
=========================================

Selection: (or <Enter> to exit) 2

  >>> Define Spatial Order Definition (SOD)? (Y/N) [Y]
     >>> Max. spatial order in Order Definition (SOD_MAX_S): [N/A](1..20) 3
     >>> Specify SOD by: (1)Euclidean Distance (2)User-defined ? (1..2) 1

Spatial order #1: Euclidean distance = [1.00,?)(1..441) 4
Spatial order #2: Euclidean distance = [4.00,?)(1..441) 9
Spatial order #3: Euclidean distance = [9.00,?)(1..441) 16

Are they good? (Y/N) [Y]

[ Spatial Order Definition ]

 0  3  3  3  3  3  0
 3  2  2  2  2  2  3
 3  2  1  1  1  2  3
 3  2  1  0  1  2  3
 3  2  1  1  1  2  3
 3  2  2  2  2  2  3
 0  3  3  3  3  3  0

  >>> Define Spatial Relation Matrix (SRM)? (Y/N) [Y]
     >>> Spatial Relation Matrix: (1)Isotropic (2)User-defined ? (1..2) 1

###          ... Isotropic structure are chosen.

[ Spatial Relation Matrix ] (SOD_MAX_S=3)

.000 .050 .050 .050 .050 .050 .000
.050 .063 .063 .063 .063 .063 .050
.050 .063 .125 .125 .125 .063 .050
.050 .063 .125 .000 .125 .063 .050
.050 .063 .125 .125 .125 .063 .050
.050 .063 .063 .063 .063 .063 .050
.000 .050 .050 .050 .050 .050 .000

Translating SRM -> SWM...

? Spatial dimension: NxN, N = [N/A] (4..100) 10

done.

Save spatial correlation structure (SOD, SRM, and SWM) to file .wet? (Y/N) [N] y

Available .wet files...
--------------------------------------
UF-S10-10x10NBPP.wet
UF-S10-16x16NBPP.wet
UF-S10-20x20NBPP.wet
UF-S6-14x14NBPP.wet

Save them as (file_name.wet)? demo.wet
```

4.6 Load Spatial Correlation Structure or Weight Matrices Directly from Files

It is easy to load an existing spatial weighting definitions from file. The three matrices *SOD*, *SRM* and *SWM* will all be loaded, and current values of them in computer memory are overwritten. The new *SOD* and *SRM* will be shown after loading. There are some pre-existing definition files for common spatial correlation structures. All of them are named in a style like: `UF-S10-10x10NBPP.wet`. `UF` means the spatial correlation is uniform or directionless. `S10` means the structure in this file can support up to spatial order (or lag) 10. `10x10` is the spatial dimension (in number of cells) supported. `NB` means spatial boundary effect is reduced. Finally, `PP` is a common option used for the configuration of correlation structure (e.g. using bishops' and rooks' moves as spatial order 1 and 2, like the one used in Section 1.2). The other option for this is `RN` which means the spatial correlation structure is defined as rings outward while the spatial order increasing (an `RN` correlation structure is used in the following example). Here is an example for loading spatial correlation structures.

```
============== [ Setup ] ==============
 [ 1] > Space-time dataset
 [ 2] > Spatial correlation structure
 [ 3] > Information of datasets
 [ 4] > Return
=======================================

Selection: (or <Enter> to exit) 2

==== [ Spatial correlation structure ] ====
 [ 1] >> Load from file (.wet)
 [ 2] >> Define by users
 [ 3] >> Return
============================================

Selection: (or <Enter> to exit) 1

Spatial correlation structure files in current directory...
------------------------------------------------------------
(1)  demo.wet
(2)  UF-S10-10x10NBPP.wet
(3)  UF-S10-16x16NBPP.wet
(4)  UF-S10-20x20NBPP.wet
(5)  UF-S6-14x14NBPP.wet
------------------------------------------------------------

Which Spatial correlation structure file to load? (Select a number): (1..5) 1

Loading spatial definition file "demo.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]

 0  3  3  3  3  3  0
 3  2  2  2  2  2  3
 3  2  1  1  1  2  3
 3  2  1  0  1  2  3
 3  2  1  1  1  2  3
 3  2  2  2  2  2  3
 0  3  3  3  3  3  0
```

```
[ Spatial Relation Matrix ] (SOD_MAX_S=3)

.000 .050 .050 .050 .050 .050 .000
.050 .063 .063 .063 .063 .063 .050
.050 .063 .125 .125 .125 .063 .050
.050 .063 .125 .000 .125 .063 .050
.050 .063 .125 .125 .125 .063 .050
.050 .063 .063 .063 .063 .063 .050
.000 .050 .050 .050 .050 .050 .000
```

Chapter 5

Data Preprocessing

To generalize STARMA modeling or to make datasets to satisfy stationarity conditions, there are various methods used for preprocessing dataset before analysis.

5.1 Remove Mean

It says everything.

5.2 De-Seasonalize

In weekly or monthly data, the seasonal component, often referred to as *seasonality*, is a component of variation in a space-time series which is dependent on the time of year. It describes any regular fluctuations with a period of less than one year. For example, unemployment figures and average daily rainfall, they all show marked seasonal variations. We are interested in comparing the seasonal effects within the years, from year to year; removing seasonal effects so that the series is easier to cope with; and, also interested in adjusting a series for seasonal effects using various models. Especially, the seasonal components make the datasets nonstationary, thereby, violate the statistical assumptions. Hence, these components needs to be removed before analysis. In IEAST, two methods for de-seasonalization are provided. Further, a spectrum analysis is provided, i.e. the strongest components or fundamental harmonics in frequency spectrum are calculated for assisting user to find the period of seasonal components. The two methods and example are shown below.

(1-B^{dd})Z(t) The de-seasonalized dataset $\mathbf{Z}_d(t) = \mathbf{Z}(t) - \mathbf{Z}(t-dd)$. dd is the period of the seasonal component.

Z(t)-$\overline{\mathbf{Z}(t)}$ Average of $\mathbf{Z}(t)$ over space is subtracted from $\mathbf{Z}(t)$ to remove seasonal component.

The following is an example of NOAA precipitation data with space=14x14 and time=120(months). We de-seasonalize it with the first method.

```
[ File name:            NA_PRCP.dat     ]
[ Data size in space:   14x14           ]
```

```
[           in time:    120             ]
[ # of Missing Data:    0               ]

========= [ Data Preprocessing ] =========
 [ 1] > Remove Mean
 [ 2] > De-seasonalize: (1-B^dd)Z(t)
 [ 3] > Diference by one: (1-B)Z(t)
 [ 4] > De-trend
 [ 5] > ------
 [ 6] > Subsequencing/Resampling
 [ 7] > Smoothing
 [ 8] > Missing Data
 [ 9] > Filter with a given STARMA model
 [10] > Undo previous action
 [11] > Return
==========================================

Selection: (or <Enter> to exit) 1

### Mean of Z has been removed.

========= [ Data Preprocessing ] =========
 [ 1] > Remove Mean
 [ 2] > De-seasonalize: (1-B^dd)Z(t)
 [ 3] > Diference by one: (1-B)Z(t)
 [ 4] > De-trend
 [ 5] > ------
 [ 6] > Subsequencing/Resampling
 [ 7] > Smoothing
 [ 8] > Missing Data
 [ 9] > Filter with a given STARMA model
 [10] > Undo previous action
 [11] > Return
==========================================

Selection: (or <Enter> to exit) 2

==== [ De-Seasonalization ] ====
  [ 1] >> (1-B^dd)Z(t)
  [ 2] >> Z(t)-Average(Z(t))
  [ 3] >> Return
================================

Selection: (or <Enter> to exit) 1
Based on spectrum of Z, the first 10 significant harmonics are at (in 1/freq):

ans =

   12.00    1.09    6.00    1.20    24.00    1.04    120.00    1.01    1.14    8.00

Deseasonalization: (1-B^dd)Z(t)

Please enter the seasonal factor to be removed. dd = (1..100) 12


### The dataset was de-seasonalized by (1-B^12)Z.

### The # of samples in time has been decreased to T=108.
```

5.3 Difference by One

Differencing is a popular and effective method of removing polynomial trends or gradients from a space-time series.

$$\mathbf{Z}_d(t) = (1 - B)\mathbf{Z}(t) = \mathbf{Z}(t) - \mathbf{Z}(t-1)$$

where B is a temporal backward operator.

5.4 De-Trend

Trend is a long term or large area movement in a space-time series. It is the underlying direction (an upward or downward tendency) and rate of change in a series, when allowance has been made for the other components. For example, in most geographical datasets, population clustering is a major trend in space and population increase is a major trend in time, for some analysis we need to remove these trends with some methods before analysis. In menu `[ De-Trend ]`, both spatial trend (trend surface) and temporal trend (trend curve) can be estimated in three different methods. The methods are shown below.

Polynomial Regression In space (2D) or in time, the trend(surface or curve) is approximated by a polynomial with give degree using ordinary least square. For example, the spatial polynomial regression trend of degree m can be expressed as follow.

$$S_{trend}(x, y) = \sum_{i=0}^{m} \sum_{j=0}^{i} a_{ij} x^j y^{i-j} \tag{5.1}$$

where a_{ij} are the best-fit coefficients.

Fourier Analysis In space (2D) or in time, the trend(surface or curve) is approximated by passing the dataset through a band-passed filter.

Average Over Space/Time Use average over space/time as the temporal/spatial trend, respectively.

Remove Trends Remove either spatial or temporal trend from dataset with various methods. Five methods are for removing spatial/temporal trends and described below:

- *Only Spatial Trend*: The same spatial trend is removed from all time period. The detrended dataset can be express as

$$Z^*_{x,y}(t) = Z_{x,y}(t) - S_{trend}(x, y).$$

- *Only Temporal Trend*: The same temporal trend is removed from all spatial sites. The detrended dataset can be express as

$$Z^*_{x,y}(t) = Z_{x,y}(t) - T_{trend}(t).$$

- *S*T Trend*: A proportioned (using temporal trend) spatial trend is removed from the dataset. The detrended dataset can be express as

$$Z^*_{x,y}(t) = Z_{x,y}(t) - S_{trend}(x, y) \cdot T_{trend}(t).$$

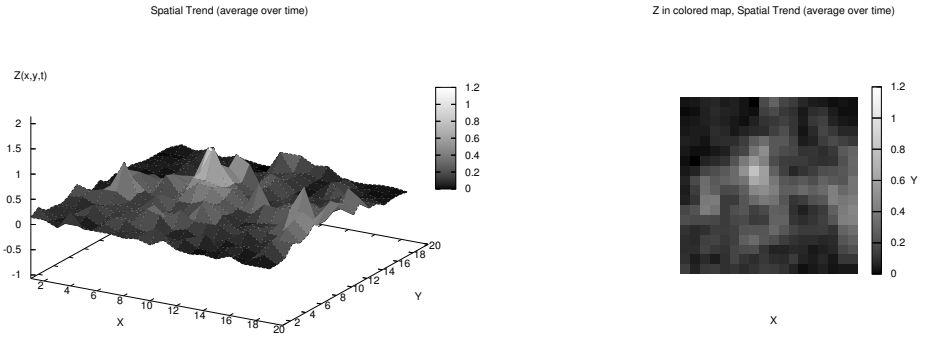

Figure 5.1: 3D plots of the original dataset.

- *Best-Fit S Trend*: A best-fit (to the dataset) factor h is found by OLS, then subtract the product of h and the spatial trend from the dataset. The detrended dataset can be express as
$$Z^*_{x,y}(t) = Z_{x,y}(t) - h \cdot S_{trend}(x,y).$$
- *Best-Fit S*T Trend*: A best-fit (to the dataset) and proportioned (using temporal trend) spatial trend is removed from the dataset. The detrended dataset can be express as
$$Z^*_{x,y}(t) = Z_{x,y}(t) - h \cdot S_{trend}(x,y) \cdot T_{trend}(t)$$
where h is a constant so that $h \cdot S_{trend}(x,y) \cdot T_{trend}(t)$ best-fit to $\mathbf{Z}(t)$

The following is an example of de-trending a dataset both in time and in space. The 3D plots of the original dataset are shown in Figure 5.1.

```
[ File name: Combined_WAM_20x20full.dat ]
[ Data size in space:   20x20           ]
[           in time:    28              ]
[ # of Missing Data:    0               ]
.
.
.
========= [ Data Preprocessing ] =========
 [ 1] > Remove Mean
 [ 2] > De-seasonalize: (1-B^dd)Z(t)
 [ 3] > Diference by one: (1-B)Z(t)
 [ 4] > De-trend
 [ 5] > ------
 [ 6] > Subsequencing/Resampling
 [ 7] > Smoothing
 [ 8] > Missing Data
 [ 9] > Filter with a given STARMA model
 [10] > Undo previous action
 [11] > Return
==========================================

Selection: (or <Enter> to exit) 4

================ [ De-trending ] ================
  [ 1] >> Trend in space (Trend Surface Analysis)
```

```
  [ 2] >> Trend in time
  [ 3] >> Remove trend in space * time
  [ 4] >> ------
  [ 5] >> Return
================================================

Selection: (or <Enter> to exit) 1

=========== [ Trend in space ] =============
  [ 1] >>> Polynomial Regression (OLS)
  [ 2] >>> Fourier Analysis
  [ 3] >>> Average over time
  [ 4] >>> Remove the Trend Surface
  [ 5] >>> Return
============================================

Selection: (or <Enter> to exit) 1

Enter the order of the trend surface; 1-1st, 2-2nd, 3-3rd,...: (1..10) 5

### Spatial trend (Poly. Regression) -> 'ZTS'.
.
.
.
================ [ De-trending ] ================
  [ 1] >> Trend in space (Trend Surface Analysis)
  [ 2] >> Trend in time
  [ 3] >> Remove trend in space * time
  [ 4] >> ------
  [ 5] >> Return
================================================

Selection: (or <Enter> to exit) 2

=========== [ Trend in time ] =============
  [ 1] >>> Polynomial Regression (OLS)
  [ 2] >>> Fourier Analysis
  [ 3] >>> Average over space
  [ 4] >>> Remove the trend
  [ 5] >>> Return
==========================================

Selection: (or <Enter> to exit) 3

### Temporal trend (Average over space) -> 'ZTC'.
.
.
.
================ [ De-trending ] ================
  [ 1] >> Trend in space (Trend Surface Analysis)
  [ 2] >> Trend in time
  [ 3] >> Remove trend in space * time
  [ 4] >> ------
  [ 5] >> Return
================================================

Selection: (or <Enter> to exit) 3

### Best-Fit Space*Time trend was removed from dataset Z(t).
```

The fifth order polynomial spatial trend and temporal trend (average over space) are shown in Figure 5.2. Finally, the dataset after de-trended both in space and in time is in Figure 5.3.

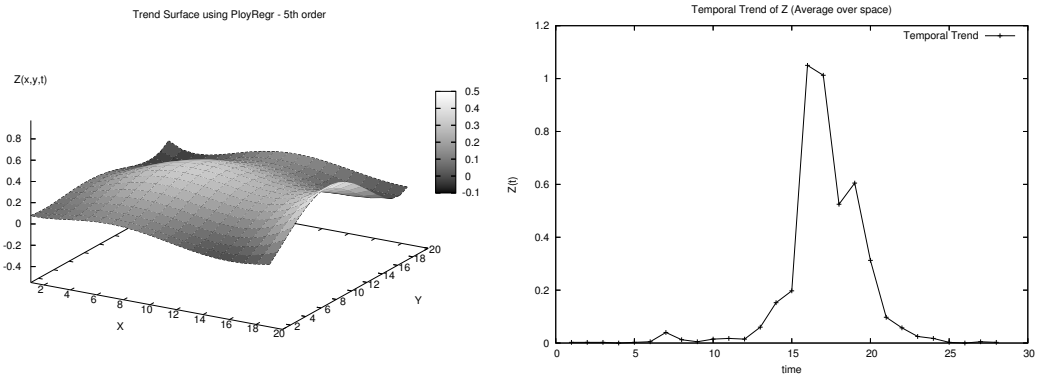

Figure 5.2: 5th order polynomial spatial trend and temporal trend (average over space).

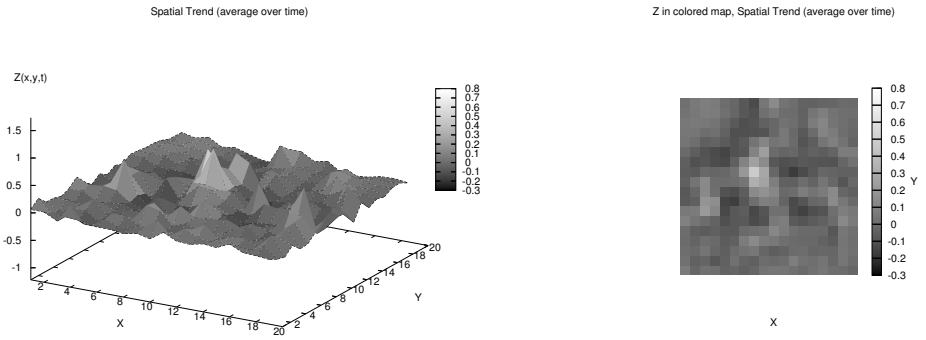

Figure 5.3: 3D plots of the dataset after de-trending.

5.5 Box-Cox Transform

Non-normality is common to all natural processes. One strategy to make non-normal data resemble normal data is by using a transformation. In menu `[ Box-Cox Transform ]`, this ability is offered. "Box-Cox transform" is a comprehensive name of a class of variance-stabilizing transforms. The transforms are given by

$$x(\lambda) = \begin{cases} \frac{x^\lambda - 1}{\lambda}, & \lambda \neq 0 \\ ln(x), & \lambda = 0. \end{cases}$$

Given a vector of observation data $\mathbf{x} = [x_1, x_2, \ldots, x_n]'$, one way to select the power λ is to use the λ that can maximize the following likelihood function

$$f(\mathbf{x}, \lambda) = -\frac{n}{2} ln\{\sum_{i=1}^{n} \frac{(x_i(\lambda) - \overline{x}(\lambda))^2}{n}\} + (\lambda - 1)\sum_{i=1}^{n} ln(x_i)$$

where

$$\overline{\mathbf{x}}(\lambda) = \frac{1}{n}\sum_{i=1}^{n} x_i(\lambda).$$

5.6 Subsequencing/Resampling

Sometimes we need the dataset to be trimmed into smaller piece either in time or in space, or resample the dataset to downscale or upscale in time or in space. In the menu `[ Subsequencing/Resampling ]`, IEAST provides the following Subsequencing/Resampling abilities.

Subsequencing in time A subset of dataset in any period of time can be picked from the original dataset.

Subsequencing in space A subset of dataset in any square area in space can be picked from the original dataset.

Resampling in time (merge) There are six methods for resampling(cell merging) the dataset in time, i.e. sum, average, number of non-zero cells, maximum, minimum and variance.

Resampling in space (merge) Same as in `[ Resampling in time (merge) ]`.

Limiting the values of in Z This is used to limit the cell values of dataset to certain extent.

5.7 Smoothing

Smoothing techniques are used to reduce irregularities (random fluctuations) in space-time series. They provide a clearer view of the true underlying behavior of a series. For example, in some series, seasonal variation is so strong it obscures any trends or cycles which are very important for the understanding of the process being observed. Smoothing can remove seasonality and makes long term fluctuations in the series stand out more clearly. The most common type of smoothing technique is moving average smoothing. Since the type of irregularities varies from series to series, so does the type of smoothing. In IEAST, both temporal and spatial smoothing are provided.

Moving Averages in space The are two different methods in moving average. They are uniform and IDM (Inverse-Distance Method). In IDM, the importance of each cell is inversely proportional (three options: $\frac{1}{d}$, $\frac{1}{d^2}$, $\frac{1}{d^3}$) to the distance between the center and the cell. An example with d^2 IDM smoothing with 3x3 window is shown below.

Moving Averages in time Not implemented yet.

Splines in space Not implemented yet.

Krigging in space Not implemented yet.

```
========= [ Data Preprocessing ] =========
 [ 1] > Remove Mean
 [ 2] > De-seasonalize: (1-B^dd)Z(t)
 [ 3] > Diference by one: (1-B)Z(t)
 [ 4] > De-trend
 [ 5] > ------
 [ 6] > Subsequencing/Resampling
 [ 7] > Smoothing
 [ 8] > Missing Data
 [ 9] > Filter with a given STARMA model
 [10] > Undo previous action
```

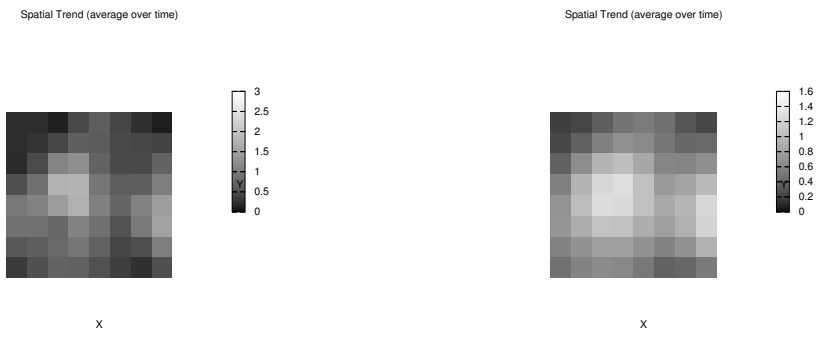

Figure 5.4: Dataset before (left) and after (right) IDM (d^2) smoothing with 3x3 window.

```
 [11] > Return
========================================

Selection: (or <Enter> to exit) 7

========== [ Smoothing ] ==========
 [ 1] >> Moving Averages in space
 [ 2] >> ------
 [ 3] >> ------
 [ 4] >> ------
 [ 5] >> Return
==================================

Selection: (or <Enter> to exit) 1

=========== [ Moving Averages ] ===========
 [ 1] >>> Uniform
 [ 2] >>> Inverse-Distance Method -- 1/d
 [ 3] >>> Inverse-Distance Method -- 1/d^2
 [ 4] >>> Inverse-Distance Method -- 1/d^3
 [ 5] >>> Return
=========================================

Selection: (or <Enter> to exit) 3

Window size in X:(odd number) (3..10) 3

Window size in Y:(odd number) (3..10) 3

done.
```

5.8 Missing Data

A critical assumption in all space-time series models is that the observations are sampled with the same frequency or distance. Unfortunately it is often the case that some values of the variable of interest are either missing or unavailable for certain dates or some locations in the sample period. There are several ways of dealing with this problem, including aggregation and interpolation.

In IEAST, the method, `[ Moving Averages in space ]`, is used to provide recovering of

missing values.

5.9 Filter with a given STARMA model

Sometimes a STARMA model found is used as a filter to eliminate the deterministic component (contribution from the STARMA process) in a dataset. The objective is to find the residuals after elimination of a given STARMA process. In this function, the deterministic component can be removed from the major variable **Z** for a STARMA model with given type, maximum spatial/temporal orders, and parameters. The following is an example of filtering the input dataset with the model

$$\mathbf{Z}(t) = 0.26\mathbf{Z}(t-1) + 0.36\mathbf{W}^{(1)}\mathbf{Z}(t-1) + 0.04\mathbf{Z}(t-2) - 0.18\mathbf{W}^{(1)}\mathbf{Z}(t-2) + \varepsilon(t)$$

and the autocorrelation functions before and after removing.

```
"WNV_XCrow_Detroit_10x10R45.dat" is being loaded ...done.

[ File name:             WNV_XCrow_Detroit_10x10R45.dat ]
[ Data size:                    Space=10x10, Time=28    ]
[ Maximum:                      27.000000               ]
[ Minimum:                      0.000000                ]
[ Sample Mean:                  0.692143                ]
[ Sample Variance:              1.545331                ]
[ Value for data missing:       -99.900000              ]
[ # of Missing Data:            0                       ]
                          .
                          .
                          .
Loading spatial definition file "UF-S10-10x10NBPP.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]

 0  0  0 10  9  9  9 10  0  0  0
 0 10  9  8  7  7  7  8  9 10  0
 0  9  8  6  5  5  5  6  8  9  0
10  8  6  5  4  3  4  5  6  8 10
 9  7  5  4  2  1  2  4  5  7  9
 9  7  5  3  1  0  1  3  5  7  9
 9  7  5  4  2  1  2  4  5  7  9
10  8  6  5  4  3  4  5  6  8 10
 0  9  8  6  5  5  5  6  8  9  0
 0 10  9  8  7  7  7  8  9 10  0
 0  0  0 10  9  9  9 10  0  0  0
                          .
                          .
                          .
========= [ Data Preprocessing ] =========
 [ 1] > Remove Mean
 [ 2] > De-seasonalize: (1-B^dd)Z(t)
 [ 3] > Diference by one: (1-B)Z(t)
 [ 4] > De-trend
 [ 5] > ------
 [ 6] > Subsequencing/Resampling
 [ 7] > Smoothing
 [ 8] > Missing Data
```

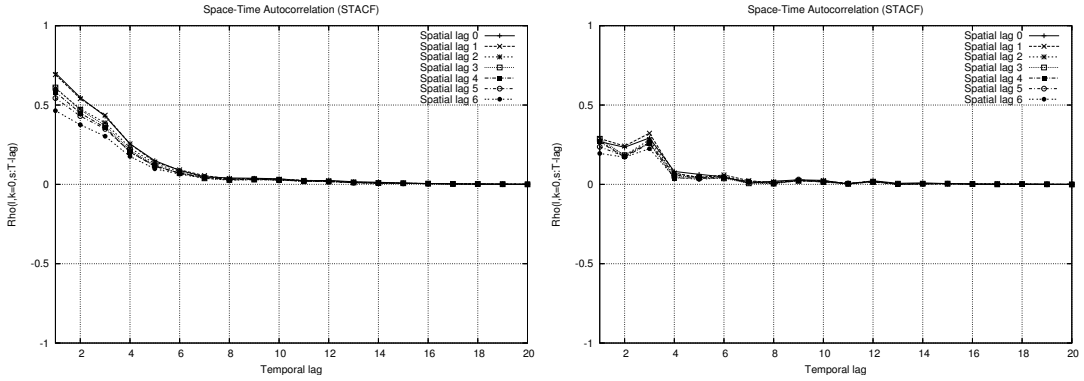

Figure 5.5: Autocorrelation functions of a dataset before (left) and after (right) filtering.

```
 [ 9] > Filter with a given STARMA model
 [10] > Undo previous action
 [11] > Return
=========================================

Selection: (or <Enter> to exit) 9
Give a STARMA model to filter the major variable Z.

  Select model type [or <Enter> for STAR]:
    [1] STAR
    [2] STMA
    [3] STARMA
  Selection: (1..3) 1

? Max. spatial order: (0..9) 1
? Max. temporal order: (1..10) 2

Assign parameters for AR terms:

Param. for AR term (t=1,s=0) = (-1..1) .26
Param. for AR term (t=1,s=1) = (-1..1) .36

Param. for AR term (t=2,s=0) = (-1..1) .04
Param. for AR term (t=2,s=1) = (-1..1) -.18


### The residuals of filtered Z is stored back to itself.
```

Chapter 6

Correlation Analyses

6.1 The Role of STACF/STPACF in STARMA Modeling

STACF and STPACF (calculated as shown in Eq. 2.5-2.7) are quite useful in identifying space-time models. There are significant differences of the STACF and STPACF among pure STAR, pure STMA, and mixed STARMA models. A tentative model can be decided by determining whether the curves of STACF or STPACF drop abruptly, or both of them decay in an exponential manner along the time (or spatial) axis. We call the abruptly dropping behavior "cut off" and the exponentially decaying behavior "tail off". By investigating at which lag (in space or in time) the STACF or STPACF cuts off, the spatial and temporal order of the model can be decided.

In a manner analogous to that of the univariate ARMA, subclasses of STARMA family are each characterized by distinct STACF and STPACF behaviors. A pure STAR model exhibits an STACF that tails off both in space and in time, and STPACF that cuts off after some lags in space and in time. In contrast, a pure STMA model is characterized by the STACF that cuts off after some temporal and spatial lags and the STPACF that tails off spatially and temporally. As to the mixed model STARMA, it exhibits STACF and STPACF that both tail off.

As examples to illustrate the cut-off/tail-off behaviors of STACF / STPACF of the dataset, we illustrate the space-time autocorrelation functions for the following two typical simulated datasets. These datasets were generated by picking the last 100 generations from a total of 5000 simulated generations. The two-dimensional space was assumed to be eight-by-eight, i.e. 64 sites in total. Uniform weight matrices, defined in [10] and [16], were used for all spatial orders (from zero to third). The random noise vector was $\boldsymbol{\varepsilon}(t) \sim N(0, \boldsymbol{\sigma}^2)$ with variance $\boldsymbol{\sigma}^2 = 0.00125\mathbf{I}$.

Example 1 The dataset in this example was generated from a pure STAR model with a maximum temporal order two and maximum spatial order one, i.e. STAR(2,1), as the following equation

$$\mathbf{Z}(t) = 0.5\mathbf{Z}(t-1) + 0.3\mathbf{W}^{(1)}\mathbf{Z}(t-1) + 0.15\mathbf{Z}(t-2) + \boldsymbol{\varepsilon}(t).$$

As the autocorrelation analysis shown in Figure 6.1, the STACF decays slowly both in space or in time, in an essentially exponential manner. In contrast, the STPACF drops abruptly at temporal lag two and spatial lag one. After these lags, the STPACF fluctuates around zero with small random disturbances.

Example 2 This dataset was generated from a pure STMA model with the maximum temporal

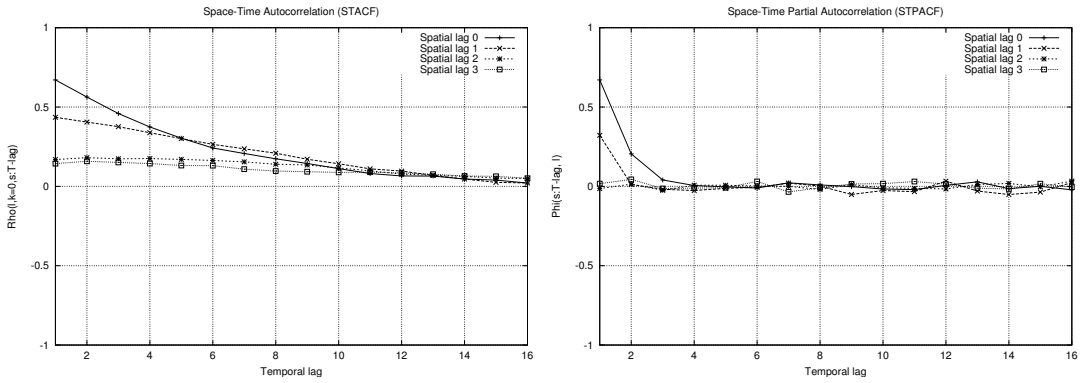

Figure 6.1: STACF and STPACF for STAR(2,1)

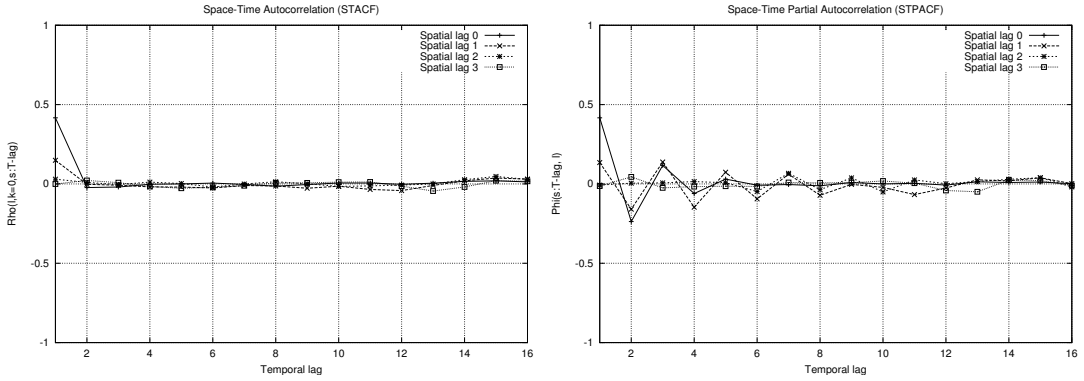

Figure 6.2: STACF and STPACF for STMA(1,1)

order one and maximum spatial order one, i.e. STMA(1,1), as the following equation

$$\mathbf{Z}(t) = \varepsilon(t) - (-0.6)\varepsilon(t-1) - (-0.4)\mathbf{W}^{(1)}\varepsilon(t-1).$$

In the results shown in Figure 6.2, instead of STACF being decaying, it drops abruptly or cuts off at both temporal lag 1 and spatial lag 1. After those lags, STACF is always near zero with small random disturbances. As to STPACF, the envelope of it decays slowly, though fluctuating markedly.

In summary, for STAR(2,1), the STPACF cuts off at temporal lag two and at spatial lag one; STPACF tails off. On the other hand, for STMA(1,1), the STACF cuts off at temporal lag one and at spatial lag one and the envelope for STPACF tails off. These two examples illustrated how the model type is selected and how the temporal/spatial orders are decided by observing the behaviors of STACF and STPACF.

6.2 Autocorrelations: STACF and STPACF

Under the menu `[ AutoCorrelation Analysis ]`, the two selections below provide the calculations of autocorrelations.

Autocorrelation (STACF), ρ_{kl} Before computing STACF or STPACF, the Spatial Weighting Structure must be specified. The spatial dimension of the weighting structure has to coincide with that of the dataset. The numbers of maximum temporal and spatial lags will be asked before calculation. The maximum temporal lag (*MAX_T_LAG*) cannot exceed the number of time periods (*T*) in the dataset. The maximum spatial lag (*MAX_S_LAG*) cannot exceed the highest spatial order (*SOD_MAX_S*) in the definition of Spatial Weighting Structure.

```
"WNV_XCrow_Detroit_10x10R45.dat" is being loaded ...done.

[ File name:              WNV_XCrow_Detroit_10x10R45.dat ]
[ Data size:                    Space=10x10, Time=28    ]
[ Maximum:                      27.000000               ]
[ Minimum:                      0.000000                ]
[ Sample Mean:                  0.692143                ]
[ Sample Variance:              1.545331                ]
[ Value for data missing:       -99.900000              ]
[ # of Missing Data:            0                       ]
                              .
                              .
                              .
Loading spatial definition file "UF-S10-10x10NBPP.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]

 0  0  0 10  9  9  9 10  0  0  0
 0 10  9  8  7  7  7  8  9 10  0
 0  9  8  6  5  5  5  6  8  9  0
10  8  6  5  4  3  4  5  6  8 10
 9  7  5  4  2  1  2  4  5  7  9
 9  7  5  3  1  0  1  3  5  7  9
 9  7  5  4  2  1  2  4  5  7  9
10  8  6  5  4  3  4  5  6  8 10
 0  9  8  6  5  5  5  6  8  9  0
 0 10  9  8  7  7  7  8  9 10  0
 0  0  0 10  9  9  9 10  0  0  0

[ Spatial Relation Matrix ] (SOD_MAX_S=10)

.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.050 .083 .063 .250 .250 .000 .250 .250 .063 .083 .050
.050 .083 .063 .125 .250 .250 .250 .125 .063 .083 .050
.083 .083 .125 .063 .125 .250 .125 .063 .125 .083 .083
.000 .050 .083 .125 .063 .063 .063 .125 .083 .050 .000
.000 .083 .050 .083 .083 .083 .083 .083 .050 .083 .000
.000 .000 .000 .083 .050 .050 .050 .083 .000 .000 .000
                              .
                              .
                              .
========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
```

```
 [ 7] > Return
================================================

Selection: (or <Enter> to exit) 1

? Remove mean of the variable Z (=0.6921428571) for autocorrelation? (Y/N) [Y]
### Mean has been removed for calculating autocorrelations.

? Max. spatial lag: [4] (0..10) 10

? Max. temporal lag: [20] (1..27) 10

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Autocorrelation of Z, Rho(s, l, k=0)
--------------------------------------------------------------------------
 S-lag  0     1     2     3     4     5     6     7     8     9     10
T-lag----------------------------------------------------------------------
  1   .669  .651  .552  .558  .522  .475  .383  .371  .291  .280  .205
  2   .493  .473  .399  .390  .361  .333  .272  .252  .194  .178  .127
  3   .357  .347  .291  .269  .245  .228  .179  .160  .117  .105  .059
  4   .159  .136  .105  .084  .057  .051  .023  .008 -.013 -.024 -.044
  5   .035 -.002 -.019 -.034 -.053 -.070 -.077 -.087 -.091 -.102 -.106
  6  -.041 -.069 -.083 -.102 -.118 -.133 -.128 -.139 -.140 -.153 -.143
  7  -.093 -.119 -.132 -.148 -.159 -.177 -.169 -.178 -.175 -.189 -.177
  8  -.109 -.151 -.154 -.171 -.182 -.204 -.192 -.202 -.195 -.212 -.194
  9  -.114 -.153 -.160 -.171 -.185 -.201 -.190 -.204 -.196 -.214 -.198
 10  -.110 -.147 -.152 -.163 -.178 -.196 -.177 -.198 -.186 -.208 -.188
```

Partial Autocorrelation (STPACF), ϕ_{kl} Similar to `[AutoCorrelation (STACF)]`. Continue on last example, the following is the result for STPACF.

```
========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 2

? Remove mean of the variable Z (=0.6921428571) for autocorrelation? (Y/N) [Y]
### Mean has been removed for calculating autocorrelations.

? Max. spatial lag: [4] (0..10) 10

? Max. temporal lag: [20] (1..27) 10

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STPACF...
```

```
|--------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Partial Autocorrelation of Z, Phi(s,l)
--------------------------------------------------------------------------
 S-lag  0     1     2     3     4     5     6     7     8     9     10
T-lag-------------------------------------------------------------------
  1    .669  .400  .037  .053 -.114 -.118  .002  .018 -.059 -.033 -.017
  2    .084 -.228 -.127 -.147 -.141  .056  .072  .035  .001 -.020  .015
  3   -.002 -.025  .007 -.058 -.046  .056 -.028 -.058 -.066 -.020 -.039
  4   -.183 -.323 -.108 -.097 -.118  .001 -.035 -.050  .024  .024  .030
  5   -.056  .023  .081  .072  .122 -.050 -.030  .010  .009  .031 -.011
  6   -.018  .005 -.036 -.060 -.047 -.118 -.000 -.058 -.023 -.073 -.000
  7   -.006  .075  .013  .007  .066  .049 -.015  .055 -.010 -.046 -.055
  8   -.005 -.119 -.061 -.070 -.102 -.202 -.025 -.128 -.055 -.107 -.038
  9   -.021 -.030  .001  .020  .032  .193  .025  .178  .092  .261  .132
 10   -.021 -.040 -.044 -.064 -.140 -.379 -.068 -.207  .386  .405  .209
```

6.3 Cross Correlations: STXCF, STPXCF, and ExtSTXCF

IEAST provides not only single variable's correlation analysis, but also the cross correlation between two space-time variables (i.e. $\mathbf{Z}$ and $\mathbf{Y}$). The model based on for this purpose is a Space-Time Regression model ([9]) and shown below.

$$\mathbf{Z}_t = \sum_{s=0}^{c} \sum_{k=1}^{d} \omega_{sk} L^s \mathbf{Y}_{t-k} + \varepsilon_t \tag{6.1}$$

where L is a spatial lag operator, and $L^s \mathbf{Z}_t = \mathbf{W}^{(s)} \mathbf{Z}_t$. The cross correlation of Z and Y is defined in [9] and shown below

$$\rho_{ZY}(s,k) = \frac{E[(Z_{it} - \mu_Z)(L^s Y_{it-k} - \mu_Y)]}{\sqrt{E[(Z_{it} - \mu_Z)^2] E[(L^s Y_{it} - \mu_Y)^2]}}. \tag{6.2}$$

In cross correlation (XCF) analysis, these two variables are ordered, that is, XCF(Z,Y) is not the same as XCF(Y,Z). XCF gives the cross correlation of the major variable $\mathbf{Z}$ and the secondary variable $\mathbf{Y}$. The temporal lag in the correlation results means how many time periods $\mathbf{Z}$ lagging behind $\mathbf{Y}$. There are three cross correlation provided in IEAST, i.e. Cross Correlation (STXCF), Partial Cross Correlation (STPXCF), and Extended Cross Correlation (ExtSTXCF). The first two are easy to understand without further explanations. As to ExtSTXCF, it calculates correlations not only on positive temporal lags, but also on negative temporal lags at the same time. Let's see the following example. The graphical result of the extended cross correlation is shown in Figure 6.3.

```
=================== [ Space-time dataset ] ===================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
 [ 6] >> Return
=============================================================

Selection: (or <Enter> to exit) 1

Dataset files in current directory...
```

```
-----------------------------------
(1) WNV_XCrow_Detroit_10x10R45.dat
(2) WNV_XHuman_Detroit_10x10R45.dat
-----------------------------------

Which Dataset file to load? (Select a number): (1..2) 2

"WNV_XHuman_Detroit_10x10R45.dat" is being loaded ...done.

[ File name:            WNV_XHuman_Detroit_10x10R45.dat ]
[ Data size:                    Space=10x10, Time=28    ]
[ Maximum:                      9.000000                ]
[ Minimum:                      0.000000                ]
[ Sample Mean:                  0.137500                ]
[ Sample Variance:              0.351836                ]
[ Value for data missing:       -99.900000              ]
[ # of Missing Data:            0                       ]

Press any key to continue...

=================== [ Space-time dataset ] ===================
 [ 1] >> Load major varialbe (Z) from dataset file (.dat)
 [ 2] >> Load second varialbe (Y) from dataset file (.dat)
 [ 3] >> Import major variable (Z) from XY+T/T+XY/XYTD
 [ 4] >> Simulation dataset
 [ 5] >> Save dataset from major variable (Z) to file (.dat)
 [ 6] >> Return
==============================================================

Selection: (or <Enter> to exit) 2

Dataset files in current directory...
-----------------------------------
(1) WNV_XCrow_Detroit_10x10R45.dat
(2) WNV_XHuman_Detroit_10x10R45.dat
-----------------------------------

Which Dataset file to load? (Select a number): (1..2) 1

"WNV_XCrow_Detroit_10x10R45.dat" is being loaded ...done.

[ File name:             WNV_XCrow_Detroit_10x10R45.dat ]
[ Data size:                    Space=10x10, Time=28    ]
[ Maximum:                      27.000000               ]
[ Minimum:                      0.000000                ]
[ Sample Mean:                  0.692143                ]
[ Sample Variance:              1.545331                ]
[ Value for data missing:       -99.900000              ]
[ # of Missing Data:            0                       ]
                          .
                          .
                          .
==== [ Spatial correlation structure ] ====
 [ 1] >> Load from file (.wet)
 [ 2] >> Define by users
 [ 3] >> Return
=========================================

Selection: (or <Enter> to exit) 1

Spatial correlation structure files in current directory...
-----------------------------------------------------------
```

```
(1)  UF-S10-10x10NBPP.wet
(2)  UF-S10-16x16NBPP.wet
(3)  UF-S10-20x20NBPP.wet
-----------------------------------------------------------

Which Spatial correlation structure file to load? (Select a number): (1..15) 1

Loading spatial definition file "UF-S10-10x10NBPP.wet" ...done.

The spatial dimension is: 10 x 10.

[ Spatial Order Definition ]

 0  0  0 10  9  9  9 10  0  0  0
 0 10  9  8  7  7  7  8  9 10  0
 0  9  8  6  5  5  5  6  8  9  0
10  8  6  5  4  3  4  5  6  8 10
 9  7  5  4  2  1  2  4  5  7  9
 9  7  5  3  1  0  1  3  5  7  9
 9  7  5  4  2  1  2  4  5  7  9
10  8  6  5  4  3  4  5  6  8 10
 0  9  8  6  5  5  5  6  8  9  0
 0 10  9  8  7  7  7  8  9 10  0
 0  0  0 10  9  9  9 10  0  0  0
                          .
                          .
                          .
========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 5

? Do you want to remove the mean of Z (=0.1375000000)? (Y/N) [Y] n

? Do you want to remove the mean of Y (=0.6921428571)? (Y/N) [Y] n

? Max. spatial lag: [10] (0..10)
### XMAX_S_LAG(=10) unchanged.

? Max. temporal lag: [27] (1..27) 12

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Initializing Covariance Look-up Table YCovarTB[] for 2nd variable Y....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Initializing Cross-Covariance Look-up Table XCovarTB[]....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating Extended STXCF ...
```

```
Initializing Cross-Covariance Look-up Table XCovarTB[]....
|-------------------------------------|
|>>>
```

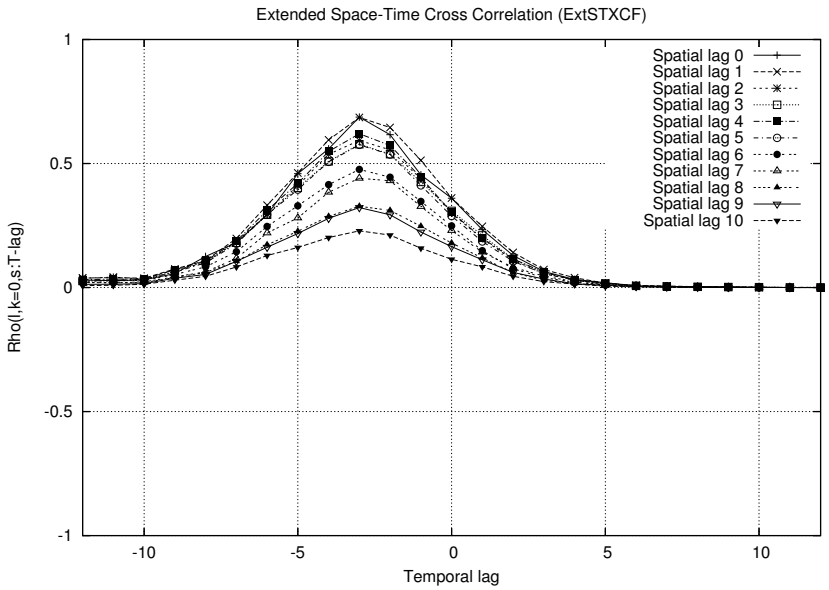

Figure 6.3: Extended cross correlations (ExtSTXCF) between two space-time variables Z and Y.

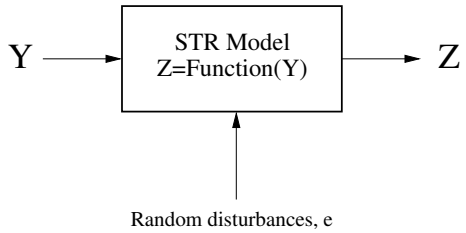

Figure 6.4: The relationship between space-time variable **Z** and independent variable **Y** under the assumption of space-time regression model.

6.4 Plotting correlations versus T-Lag/S-Lag for Z/Y

The plotting function for correlations is under the menu `[Plot STACF/STPACF/STPXCF/ExtSTXCF vs T/S-Lag]`. There are two things need to be specified before plotting.

Correlation to be plotted The correlations can be plotted are STACF, STPACF, STXCF, STPXCF, and ExtSTXCF. User need select one of them to plot.

Plot versus T-Lag or S-Lag The X-axis in the plot can be chosen to be either T-Lag or S-Lag. If T-Lag is on X-axis, then S-Lags are represented in different colored curves, and vice versa. User need to decide which to be plotted on X-axis (i.e. T-Lag or S-Lag).

As an example, we continue the example in section 6.2 and plot its result (as shown in Figure 6.5).

```
========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
```

```
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 6

  Which correlation to plot:
    [1] STACF  of Z
    [2] STPACF of Z
    [3] STACF  of Y
    [4] STPACF of Y
    [5] STXCF  of Z and Y
    [6] STPXCF of Z and Y
    [7] Extended STXCF
  Selection: (1..7) 1

? Ploting from spatial lag: [0] (0..10)
? Ploting up to spatial lag: [10] (0..10)
? Ploting from temporal lag: [1] (1..10)
? Ploting up to temporal lag: [10] (1..10)

  Plotting versus:
    [1] Temporal lags
    [2] Spatial lags
  Selection: (1..2) 1

### Plotting STACF of Z versus T-Lag...
### The plot is also saved as an eps file (ACF_PLOT_T.eps).

========== [ Correlation Analyses ] ==========
 [ 1] > AutoCorrelation (STACF)
 [ 2] > Partial AutoCorrelation (STPACF)
 [ 3] > Cross Correlation (STXCF)
 [ 4] > Partial Cross Correlation (STPXCF)
 [ 5] > Extended Cross Correlation (ExtSTXCF)
 [ 6] > Plot Correlations versus T-Lag/S-Lag
 [ 7] > Return
==============================================

Selection: (or <Enter> to exit) 6

  Which correlation to plot:
    [1] STACF  of Z
    [2] STPACF of Z
    [3] STACF  of Y
    [4] STPACF of Y
    [5] STXCF  of Z and Y
    [6] STPXCF of Z and Y
    [7] Extended STXCF
  Selection: (1..7) 1

? Ploting from spatial lag: [0] (0..10)
? Ploting up to spatial lag: [10] (0..10)
? Ploting from temporal lag: [1] (1..10)
? Ploting up to temporal lag: [10] (1..10)

  Plotting versus:
    [1] Temporal lags
```

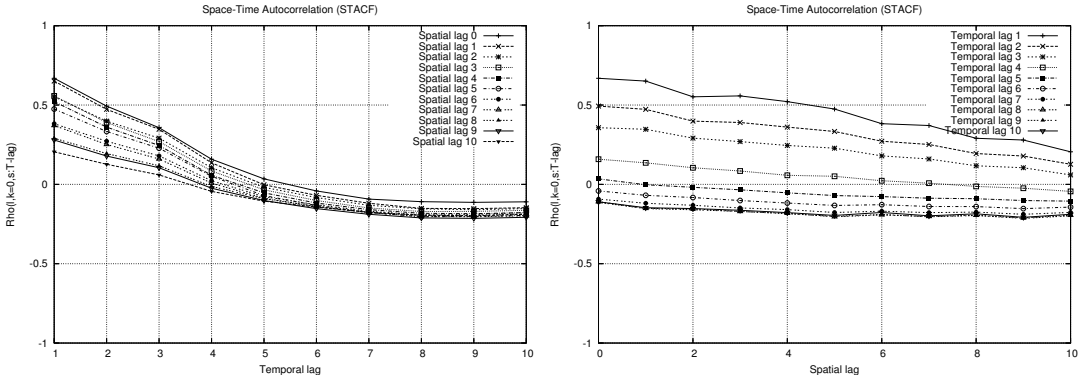

Figure 6.5: STACF plots versus T-Lag (left) and S-Lag (right)

```
    [2] Spatial lags
  Selection: (1..2) 2

### Plotting STACF of Z versus S-Lag...
### The plot is also saved as an eps file (ACF_PLOT_S.eps).
```

Chapter 7

Model Identification

Before estimation stage, we have to let IEAST know what the model type and orders would be appropriate for the underlying process so that the number of parameters to be estimated or even the estimation algorithm to be used can be decided. There are many methods can be used to determine type/orders, e.g. using statistical measures for model fitness to search appropriate model iteratively, using STACF/STPACF. In the automatic identification, IEAST uses the behavior of STACF/STPACF to decide type/orders.

7.1 Automatic Identification

Based on the cut-off/tail-off behavior of the correlation results, we can tentatively decide the type/orders of the dataset as described in section 6.1. For most of pure STAR and pure STMA processes, the type/orders can be easily identified by users. However, for the mixed STARMA and some STAR/STMA models, it is hard to distinguish them by "eyes". Therefore, IEAST provides an algorithm (select `[ Automatic Identification (Type,Orders) ]` to replace human inspection and to assist users making decision. This identification algorithm is also based on correlation analyses. Notice that this functionality of automatic identification is not necessary correct, especially for some practical datasets which have ambiguous behaviors on STACF/STPACF.

```
=========== [ Model Identification ] ===========
 [ 1] > Automatic Identification (Type,Orders)
 [ 2] > Artificial Identification (Type,Orders)
 [ 3] > Parameter Masking
 [ 4] > ------
 [ 5] > Return
================================================

Selection: (or <Enter> to exit) 1

### Model type is suggested to be STMA with
### Max. Temporal Order = 3, Max. Spatial Order = 2.
```

7.2 Artificial Identification

If users want to specify the model type/orders according to their knowledge to this dataset, select `[ Artificial Identification (Type,Orders) ]`.

```
=========== [ Model Identification ] ===========
 [ 1] > Automatic Identification (Type,Orders)
 [ 2] > Artificial Identification (Type,Orders)
 [ 3] > Parameter Masking
 [ 4] > ------
 [ 5] > Return
================================================

Selection: (or <Enter> to exit) 2

Model Type(1=STAR,2=STMA,3=STARMA)? [STMA] (1..3) 1

Max. temporal order(1..20)? [3] (1..20) 2

Max. spatial order(0..20)? [2] (0..20) 2

### Model type is assigned to be STAR with
### Max. Temporal Order = 2, Max. Spatial Order = 2.
```

7.3 Parameter Masking

If there are some parameters need to be temporarily removed or put more attention to, users can select `[ Parameter Masking ]` to eliminate the effects in estimation stage of those 'unwanted' parameters. The following is an example.

```
=========== [ Model Identification ] ===========
  [ 1] > Automatic Identification (Type,Orders)
  [ 2] > Artificial Identification (Type,Orders)
  [ 3] > Parameter Masking
  [ 4] > ------
  [ 5] > Return
================================================

Selection: (or <Enter> to exit) 1

### Model type is suggested to be STAR with
### Max. Temporal Order = 3, Max. Spatial Order = 2.

Press any key to continue...

=========== [ Model Identification ] ===========
  [ 1] > Automatic Identification (Type,Orders)
  [ 2] > Artificial Identification (Type,Orders)
  [ 3] > Parameter Masking
  [ 4] > ------
  [ 5] > Return
================================================

Selection: (or <Enter> to exit) 3

Set masks for AR terms (Y/N)[N]? y
```

```
Setting masks for AR terms (Enable/Disable)[E]

AR term (t=1,s=0)...(Enable/Disable)  (E/D)[E]
AR term (t=1,s=1)...(Enable/Disable)  (E/D)[E]
AR term (t=1,s=2)...(Enable/Disable)  (E/D)[E]
AR term (t=2,s=0)...(Enable/Disable)  (E/D)[E] d
AR term (t=2,s=1)...(Enable/Disable)  (E/D)[E]
AR term (t=2,s=2)...(Enable/Disable)  (E/D)[E] d
AR term (t=3,s=0)...(Enable/Disable)  (E/D)[E]
AR term (t=3,s=1)...(Enable/Disable)  (E/D)[E] d
AR term (t=3,s=2)...(Enable/Disable)  (E/D)[E]

Set masks for MA terms (Y/N)[N]?
```

Chapter 8

Parameter Estimation

Estimation is the most time-consuming stage in some cases. In estimation, the parameters in the candidate model with the selected type and orders are estimated. We implemented this stage with the maximum likelihood estimates(MLE). In IEAST, we have implemented not only the algorithms for estimating model parameters, but also the algorithms for spatial weighting structure. In other words, IEAST can retrieve the spatial weighting information from the dataset.

8.1 Estimation of Model Parameters

Because of nonlinear nature of STMA and STARMA, we have to use a nonlinear optimization algorithm to find the parameters for STARMA family. Marquardt's algorithm [8] was chosen to implement the nonlinear optimization of model parameters.

During the optimization process, to avoid the likelihood function reaching a local maxima and to reduce the iterations needed in the optimization, we added an extra stage, pre-estimation, to calculate the appropriate initial values for the optimization algorithm. In IEAST, there are four pre-estimation methods implemented: (1) for linear processes (STAR), (2) for non-linear processes (STMA,mixed model), (3) from STACF/STPACF values, (4) given by users. All of them are discussed in section 8.3.

8.2 Estimation of Spatial Weighting Structure

We found that, in some cases, spatial correlation structure is more important than the estimates of model parameters. Especially, spatial correlation structure can provide directional information (anisotropy) and future tendency of a process. Thus, IEAST provides several methods to discover the spatial correlation structure by estimating the spatial weighting matrices for a given dataset. There are three important matrices related to spatial correlation structure in IEAST, i.e. **SOD**, **SRM**, and **SWM**. **SWM** is in a form of computation-friendly, but big and awkward. In fact, the compact form, **SOD+SRM**, has one-by-one mapping to **SWM** and concise. Therefore, in estimation of spatial weighting structure, **SOD+SRM** are the target to estimate. Actually, only the elements in matrix **SRM** are needed to be estimated. Because the model parameters and **SRM** are in a same model equation (in some sense, at least), it is not wise to estimate all of them

at the same time. Fortunately, we found that reasonable small deviations in **SRM** will not affect the estimation of the model parameters much. Thus, while implementing these algorithms, we intentionally separated these two objectives (estimations of spatial correlation structure and model parameters). There is also an algorithm designed to estimate these two together but alternatively. In section 8.3, these algorithm are further discussed and shown.

8.3 Estimation Algorithms in IEAST

There are several estimation algorithms for different purposes. The first four algorithms are pre-estimations, i.e. estimating initial values. The fifth is for estimating model parameters only. The sixth is only for estimating spatial weighting structure in the form of SRM, that is to say, the model parameters are given and fixed. The last one is to estimate both of them, but alternatively, that is, in every iteration, the estimations of model parameters and SRM will run once.

1. **Pre-estimate Model Param – Linear (STAR)** Estimate initial values for linear processes STAR. The best linear unbiased estimator (BLUE) [12] is used to find the initial values.

2. **Pre-estimate Model Param – Non-linear (STMA,STARMA)** Estimate initial values for non-linear process STMA and STARMA. A space-time extension of Hannan-Rissanen algorithm [4] was implemented for this purpose.

3. **Pre-estimate Model Param – From STACF/STPACF** No matter what model type it is, once the type is determined, the initial values of model parameters can be retrieved from the coefficients in the first few spatial/temporal lags in STACF (or STPACF, depending on model type). This is a quicker way to get the initial values, and thereafter the algorithm in *Algorithm 5* can optimize the estimates.

4. **Pre-estimate Model Param – Specified by users** Users assign initial values based on their knowledge to the dataset.

5. **Estimate Model Param – Fixed SRM** If we know the spatial correlation structure (basically, SRM) before estimating model parameters, this algorithm is used to find the estimates of model parameters. Sometimes, even spatial correlation structure is unknown, we still can use this algorithm with the assumption of isotropic SRM. The values of elements in SRM will not affect too much to the model parameters, if we use a 'reasonable' spatial correlation structure, such as a uniform or isotropic SRM.

6. **Estimate SRM – Fixed Model Param** If the model parameters are known, then we can estimate the SRM with fixed model parameters. For example, the parameters in STAR model can simply be estimated by *Algorithm 1* which is quick. Thereafter, the SRM can be estimated using the initial estimates of model parameters.

7. **Estimate SRM & Model Param – Alternatively** With an isotropic SRM as the initial SRM, the SRM and model parameters can be estimated alternatively. When converged, we can obtain a set of model parameter estimates and a SRM estimate. However, because of so many variables in SRM to be estimated, in general, this algorithm is very time-consuming.

In summary, considering trade-offs on computing time and accuracy, we have the following suggestions:

- If the model type is STAR, then use *Algorithm 1* to estimate model parameters (this is the best estimate can be found for a linear process and there is no need to optimize it with other algorithms). If users also like to find the spatial correlation structure, then fix the model parameters and estimate SRM with *Algorithm 6*.
- If the model type is STMA, then use *Algorithm 2* to estimate initial values of model parameters. If users also like to find the spatial correlation structure, then use *Algorithm 7* to estimate both model parameters and SRM.
- If the model type is STARMA, then use *Algorithm 2* to estimate initial values of model parameters. If users also like to find the spatial correlation structure, then fixe model parameters and estimate SRM with *Algorithm 6*.

The following is the estimation stage of the example in Quick Start(section 1.2). In Quick Start, we only estimated the model parameters. Here, we also estimate the spatial weighting structure, SRM. It takes long time to get the results.

```
================== [ Parameter Estimation ] ==================
 [ 1] > Pre-estimate Model Param   -- Linear (STAR)
 [ 2] > Pre-estimate Model Param   -- Non-linear (STMA,STARMA)
 [ 3] > Pre-estimate Model Param   -- From STACF/STPACF
 [ 4] > Pre-estimate Model Param   -- Specified by users
 [ 5] > Estimate Model Param       -- Fixed SRM
 [ 6] > Estimate SRM               -- Fixed Model Param
 [ 7] > Estimate SRM & Model Param -- Alternatively
 [ 8] > Return
==============================================================

Selection: (or <Enter> to exit) 1
Phi_Now =

   0.30   0.23   0.12
  -0.20  -0.13  -0.07

Press any key to continue...

================== [ Parameter Estimation ] ==================
 [ 1] > Pre-estimate Model Param   -- Linear (STAR)
 [ 2] > Pre-estimate Model Param   -- Non-linear (STMA,STARMA)
 [ 3] > Pre-estimate Model Param   -- From STACF/STPACF
 [ 4] > Pre-estimate Model Param   -- Specified by users
 [ 5] > Estimate Model Param       -- Fixed SRM
 [ 6] > Estimate SRM               -- Fixed Model Param
 [ 7] > Estimate SRM & Model Param -- Alternatively
 [ 8] > Return
==============================================================

Selection: (or <Enter> to exit) 7

### Estimating Model Parameters & SRM alternatively...

------- Iter#1:(Fri Sep 19 10:57:56 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.25  0.25  0.25
  0.25  1.00  0.25
  0.25  0.25  0.25

ePHI =
```

```
   0.30   0.23   0.12
  -0.20  -0.13  -0.07

Advancement of Average Parameter SSE = 0.074186
Time elapsed = 1:46:43.94

------- Iter#2:(Fri Sep 19 12:44:40 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.30  0.22  0.19
  0.32  0.00  0.29
  0.18  0.28  0.39

ePHI =
   0.30   0.22   0.12
  -0.20  -0.13  -0.07

Advancement of Average Parameter SSE = 0.000422
Time elapsed = 1:43:45.69

------- Iter#3:(Fri Sep 19 14:28:26 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.31  0.23  0.20
  0.35  0.00  0.31
  0.20  0.29  0.45

ePHI =
   0.30   0.21   0.11
  -0.20  -0.12  -0.07

Advancement of Average Parameter SSE = 0.000105
Time elapsed = 1:48:5.26

------- Iter#4:(Fri Sep 19 16:16:31 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.32  0.23  0.20
  0.37  0.00  0.32
  0.20  0.31  0.47

ePHI =
   0.30   0.20   0.11
  -0.20  -0.12  -0.07

Advancement of Average Parameter SSE = 0.000045
Time elapsed = 1:46:11.13

------- Iter#5:(Fri Sep 19 18:02:42 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.33  0.24  0.21
  0.38  0.00  0.33
  0.21  0.31  0.48

ePHI =
   0.30   0.20   0.11
  -0.20  -0.12  -0.07

Advancement of Average Parameter SSE = 0.000021
Time elapsed = 1:43:50.98
```

```
------- Iter#6:(Fri Sep 19 19:46:33 2003) -------
[ Spatial Relation Matrix ] (SOD_MAX_S=6)
SRM =
  0.34  0.25  0.21
  0.38  0.00  0.33
  0.21  0.32  0.49

ePHI =
   0.30   0.20   0.11
  -0.20  -0.12  -0.07

Advancement of Average Parameter SSE = 0.000010
Time elapsed = 1:47:56.93

Z(t) =
( 0.30)*Z(t-1) + ( 0.20)*W1*Z(t-1) + ( 0.11)*W2*Z(t-1) +
(-0.20)*Z(t-2) + (-0.12)*W1*Z(t-2) + (-0.07)*W2*Z(t-2) +
e(t)
```

Chapter 9

Diagnostic Analysis

Once the model type, order and parameters have been identified and estimated, the acquired model can be used to forecast or to analyze the future behavior of the system of interest. To prevent unacceptable forecasting errors, we have to check if the fitted model is appropriate. If there is abundant data on hand, it would be beneficial to identify and estimate the model on the first half, and check adequacy of the model on the remainder. If the amount of data is limited, models can be identified, estimated, and diagnostically checked using the same dataset.

These checking methods are called diagnostic checks. There are various methods to check adequacy of STARMA models, e.g. autocorrelation checking, randomness testing, variance checking [12, 13], statistical significance testing, and statistical measures (like AICC/BIC). In IEAST, autocorrelation checking, variance checking, statistical significance testing, and AICC/BIC are provided.

9.1 STACF of Residuals

The most commonly used method for diagnostic checking is to investigate the behaviors of the STACF and STPACF of the residuals. Inadequacy of a candidate model will be unveiled in the form of significant space-time correlations among the residuals. If the candidate model adequately represents the data, residuals must be white(uncorrelated) noise, that is, all autocorrelations (in space or time) at nonzero lags are equal to zero (or approximately equal to zero, in practice). If we compute the residuals' STACF and STPACF of the example in Quick Start (section 1.2), we have the results below. The plots of residuals' STACF/STPACF are shown in Figure 9.1.

```
====== [ Diagnostic Analysis ] ======
  [ 1] > Statistical Significance
  [ 2] > AICC/BIC Analysis
  [ 3] > STACF of Residuals
  [ 4] > STPACF of Residuals
  [ 5] > ------
  [ 6] > Return
====================================

Selection: (or <Enter> to exit) 3

Initializing Covariance Look-up Table CovarTB[]....
```

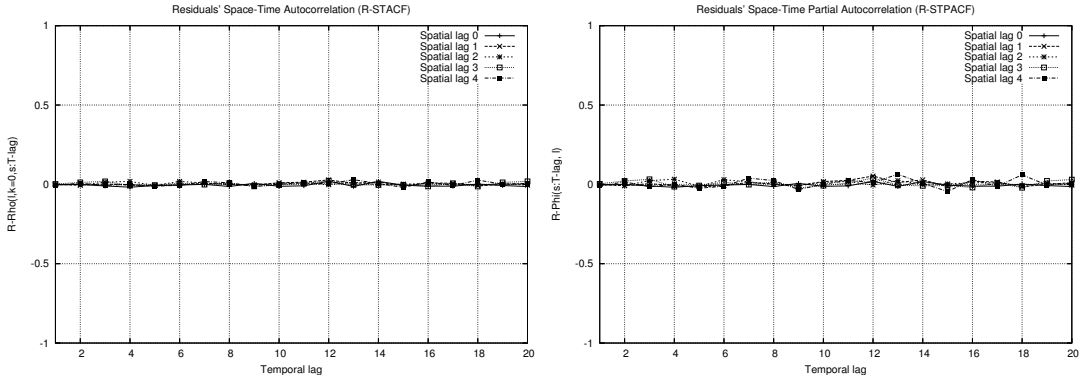

Figure 9.1: Residuals' STACF and STPACF for the example in Quick Start

```
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Autocorrelation, Rho(s, l, k=0)
------------------------------------------
   S-lag    0      1      2      3      4
T-lag  ----------------------------------
   1    -0.003 -0.001  0.001  0.002 -0.005
   2    -0.001 -0.004 -0.004  0.012  0.007
   3    -0.010  0.002  0.014  0.019 -0.007
   4    -0.018 -0.004  0.019 -0.008  0.000
   5    -0.008 -0.009 -0.007 -0.005 -0.011
   6    -0.004 -0.004  0.018  0.007 -0.006
   7     0.000  0.007  0.009 -0.001  0.018
   8    -0.013  0.001 -0.001  0.009  0.010
   9     0.007 -0.003 -0.005 -0.013 -0.014
  10    -0.012  0.010  0.007 -0.004  0.002
  11    -0.009  0.014  0.009  0.005  0.010
  12     0.019  0.028 -0.003  0.023  0.003
  13    -0.013  0.006  0.013 -0.004  0.030
  14     0.019  0.015 -0.002 -0.004  0.005
  15    -0.005 -0.008  0.002 -0.004 -0.019
  16    -0.011  0.009  0.004 -0.013  0.015
  17    -0.012  0.007  0.002  0.007 -0.008
  18     0.002 -0.009 -0.004 -0.013  0.027
  19    -0.007  0.000  0.004  0.012  0.002
  20    -0.014  0.005 -0.005  0.019  0.005
```

9.2 STPACF of Residuals

Same as `[ STACF of Residuals ]`. Continue on the last example, the residuals' STPACF is shown below. The results of STACF and STPACF show that there is no significant correlation in space and in time. That implies the selected candidate may be a good model to represent the dataset.

```
====== [ Diagnostic Analysis ] ======
  [ 1] > Statistical Significance
  [ 2] > AICC/BIC Analysis
  [ 3] > STACF of Residuals
  [ 4] > STPACF of Residuals
  [ 5] > ------
  [ 6] > Return
====================================

Selection: (or <Enter> to exit) 4

Calculating STPACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Partial Autocorrelation, Phi(s,l)
--------------------------------------------
   S-lag    0      1      2      3      4
T-lag  ----------------------------------
   1    -0.003 -0.002  0.001  0.003 -0.010
   2    -0.001 -0.008 -0.007  0.021  0.015
   3    -0.010  0.004  0.024  0.032 -0.014
   4    -0.018 -0.008  0.032 -0.014 -0.000
   5    -0.008 -0.016 -0.012 -0.008 -0.024
   6    -0.005 -0.008  0.030  0.013 -0.012
   7    -0.000  0.013  0.016 -0.002  0.040
   8    -0.014  0.002  0.000  0.015  0.024
   9     0.006 -0.006 -0.010 -0.022 -0.032
  10    -0.012  0.018  0.014 -0.008  0.001
  11    -0.009  0.026  0.018  0.008  0.025
  12     0.019  0.053 -0.004  0.040  0.011
  13    -0.013  0.011  0.024 -0.006  0.064
  14     0.018  0.029  0.001 -0.009  0.007
  15    -0.005 -0.014  0.003 -0.007 -0.045
  16    -0.011  0.019  0.011 -0.019  0.031
  17    -0.011  0.015  0.010  0.009 -0.013
  18     0.002 -0.015 -0.007 -0.021  0.060
  19    -0.007  0.001  0.009  0.021 -0.005
  20    -0.014  0.010 -0.005  0.030  0.007
```

9.3 Statistical Significance

A tentative model need to be checked to see if it is unduly complex, i.e. having statistically insignificant terms. The statistical significance of model parameters is evaluated for this purpose. Statistical significance for each estimated parameters can be calculated as below.

We test the hypothesis that a parameter of interest $\hat{\beta}_k$ is zero, keeping other parameters unrestricted. That is, let $\hat{\boldsymbol{\beta}}^* = [\hat{\beta}_1, \hat{\beta}_2, \cdots, \hat{\beta}_k = 0, \hat{\beta_{k+1}}, \cdots]'$. The appropriate test for this hypothesis is based on the statistic

$$\frac{[TN - K(\hat{\boldsymbol{\Phi}}) - K(\hat{\boldsymbol{\Theta}})][S(\hat{\boldsymbol{\beta}}^*) - S(\hat{\boldsymbol{\beta}})]}{S(\hat{\boldsymbol{\beta}})} \sim F_{1, TN-K(\hat{\boldsymbol{\Phi}})-K(\hat{\boldsymbol{\Theta}})} \tag{9.1}$$

where T is the number of points in time in the dataset, N is the total number of sites in the dataset, $S(\hat{\boldsymbol{\beta}}^*)$ is the sum of square residuals for parameter $\hat{\boldsymbol{\beta}}$ with the restriction $\hat{\beta}_k = 0$, $S(\hat{\boldsymbol{\beta}})$ is the sum of square residuals for parameter $\hat{\boldsymbol{\beta}}$ without restriction, $\hat{\boldsymbol{\Phi}}$ is the estimated parameters for autoregressive terms in column vector form, $\hat{\boldsymbol{\Theta}}$ is the estimated parameters for moving average

terms in column vector form, $K(\hat{\mathbf{\Phi}})$ is the number of elements in $\hat{\mathbf{\Phi}}$, $K(\hat{\mathbf{\Theta}})$ is the number of elements in $\hat{\mathbf{\Theta}}$. Significance for the parameter $\hat{\beta_k}$ can be found by comparing with standard tables of the F-distribution, using 1 and $TN - K(\mathbf{\Phi}) - K(\mathbf{\Theta})$ degrees of freedom.

During the modeling process, any estimated parameter proved being statistically insignificant should be removed from the model, and sometimes space/time orders should be reduced to obtain a simpler model, which is then considered as the next candidate model. Parameters with significance probability < 0.8 can be tentatively regarded as insignificant from a practical standpoint. The following is an analysis of statistical significance to the example in Quick START (section 1.2). Note: The analysis of statistical significance will work only after *Estimation* has been done.

```
====== [ Diagnostic Analysis ] ======
  [ 1] > Statistical Significance
  [ 2] > AICC/BIC Analysis
  [ 3] > STACF of Residuals
  [ 4] > STPACF of Residuals
  [ 5] > ------
  [ 6] > Return
====================================

Selection: (or <Enter> to exit) 1

--- Analysis of Statistical Significance ----

STAR Param.(t= 1,s= 0) =  0.30 (F= 1021.97) ... P=0.001
STAR Param.(t= 1,s= 1) =  0.23 (F=  181.17) ... P=0.001
STAR Param.(t= 1,s= 2) =  0.12 (F=   56.39) ... P=0.001
STAR Param.(t= 2,s= 0) = -0.20 (F=  460.46) ... P=0.001
STAR Param.(t= 2,s= 1) = -0.13 (F=   58.41) ... P=0.001
STAR Param.(t= 2,s= 2) = -0.07 (F=   20.93) ... P=0.001

Estimates & P-values for AR terms:

      S= 0 S= 1 S= 2
---------------------
T= 1   .30  .23  .12
      .001 .001 .001
---------------------
T= 2  -.02 -.13 -.07
      .001 .001 .001
---------------------
```

9.4 AICC/BIC Analysis

In addition to the analyses of residuals' autocorrelations and parameter statistical significance, some other tools can also be helpful in model determination. In many cases, the STACF and STPACF do not obviously cut off or tail off as they did in pure STAR or pure STMA models. In such situations, it is not easy to determine the type and orders by observing the behaviors of STACF and STPACF.

A STAR model with invertibility can be represented as a high order STMA model. On the other hand, a STMA model with causality can be represented as a high order STAR model. Fitting a very high order model will result in a small estimated noise variance, but it is not necessarily desirable. In fact, it is not advantageous from a forecasting point of view to choose orders arbitrarily

large. Hence, we need to introduce a penalty into the criterion to discourage the fitting of models with too many parameters.

We extended and implemented a space-time extension of bias-corrected AIC [1], referred to as the AICC, suggested by Hurvich and Tsai[6], to be the criterion in determining the model type, orders and checking adequacy of parameters.

The space-time extension of AICC criterion for STARMA can be defined as follows.

$$AICC_{STARMA} = TNlog(\hat{\sigma}^2) + \frac{TN + K(\hat{\mathbf{\Phi}}) + K(\hat{\mathbf{\Theta}})}{TN - K(\hat{\mathbf{\Phi}}) - K(\hat{\mathbf{\Theta}}) - 2} \tag{9.2}$$

where the maximum likelihood estimate of the residual variance $\hat{\sigma}^2$ is

$$\hat{\sigma}^2 = \frac{S(\hat{\mathbf{\Phi}}, \hat{\mathbf{\Theta}})}{TN},$$

the residuals' sum of square errors is

$$S(\hat{\mathbf{\Phi}}, \hat{\mathbf{\Theta}}) = \sum_{t=1}^{T} \left(\mathbf{Z}(t) - \sum_{k=1}^{p} \sum_{l=0}^{\lambda_k} \hat{\phi}_{kl} \mathbf{W}^{(l)} \mathbf{Z}(t-k) + \sum_{k=1}^{q} \sum_{l=0}^{m_k} \hat{\theta}_{kl} \mathbf{W}^{(l)} \hat{\varepsilon}(t-k) \right)^2,$$

T is the number of points in time of the observations, N is the number of sites in each observation, $\hat{\mathbf{\Phi}}$ is the estimated parameters for autoregressive terms, $\hat{\mathbf{\Theta}}$ is the estimated parameters for moving average terms, $K(\hat{\mathbf{\Phi}})$ is the number of elements in $\hat{\mathbf{\Phi}}$, $K(\hat{\mathbf{\Theta}})$ is the number of elements in $\hat{\mathbf{\Theta}}$.

In the definition of AICC (Eq. 9.2), the second term is the penalty for high order models, which makes the value of AICC higher than that of lower order model. Thus, in determining an adequate model, we only need to find a model (with some type, orders, and parameters) which has a minimum value of AICC.

However, because AICC has tendency of overfitting autoregressions as shown in [7] and [15], another criterion, i.e. BIC(Bayesian Information Criterion), is suggested to correct this tendency. BIC is suggested by [2, 14] and extended to space-time case as follows:

$$BIC_{STARMA} = TNlog(\hat{\sigma}^2) + \frac{S(\hat{\boldsymbol{\beta}})}{\hat{\sigma}^2} + (K(\hat{\mathbf{\Phi}}) + K(\hat{\mathbf{\Theta}}))log(TN). \tag{9.3}$$

Both AICC and BIC are relative measures. They are useful when select the best model out of many candidates. In IEAST, they are especially useful when using the IEAST programming environment in which iterative testing of AICC/BIC on different models is a powerful method to pick a good model for the dataset out of lots of candidates.

The following is an analysis of AICC/BIC to the example in Quick Start (section 1.2).

```
====== [ Diagnostic Analysis ] ======
  [ 1] > Statistical Significance
  [ 2] > AICC/BIC Analysis
  [ 3] > STACF of Residuals
  [ 4] > STPACF of Residuals
  [ 5] > ------
  [ 6] > Return
=====================================
```

```
Selection: (or <Enter> to exit) 2

-------------------------------------------
The residual sum of square errors = 12.70624
Estimated Mean = -0.00001
Estimated Noise Variance = 0.00127
AICC = -66675.467
BIC = -28313.603
-------------------------------------------
```

9.5 Residuals' Variance

With regard to variance checking, it has been shown by [10] that if a process (the residuals, in our case) is pure white noise, then

$$variance(\hat{\rho}_{l0}(s)) \approx \frac{1}{N(T-s)} \tag{9.4}$$

where $\hat{\rho}_{l0}(s)$ is the sample STACF as defined in Eq. 2.5. That is, if the estimated model appropriately represents the dataset, the variance of sample STACF of residuals should satisfy Eq. 9.4.

Chapter 10

Interpreter

In IEAST, the programming environment provided is an interpreter. In general, an interpreter parses and executes requested actions in every statement line by line in a program. This environment provides a method using scripts to compose a modeling job.

10.1 The Programming Environment

From the point of view of functionalities provided, interpreter is parallel to menu-driven environment. Each of them can separately work well. However, the interpreter gives an even more powerful environment for modeling STARMA processes than menu-driven environment does. Combining with the suggested STARMA modeling procedure (Figure 2.4), under the interpreter users can construct a complete and efficient modeling flow with maximum flexibility. The interactive response and attention of users are not required. For the different datasets with the same setup and modeling flow, the only thing different in the program is the instructions for retrieving dataset from different files. Users need not to program it again.

10.1.1 Invoke and Quit

To enter the interpreter, select `[ Interpreter ]` under the main menu. As we expected, there is a prompt '`#`' after entering. That means it is ready to execute commands or accept program input. To leave the interpreter, type command `quit` and press `<Enter>` key as shown below.

```
================= Main Menu ================
  [ 1] Setup
  [ 2] Data Preprocessing
  [ 3] AutoCorrelation Analysis
  [ 4] Model Identification
  [ 5] Parameter Estimation
  [ 6] Diagnostic Analysis
  [ 7] Forecasting
  [ 8] Simulation
  [ 9] Preference
  [10] Interpreter
  [11] Exit
============================================
```

```
Selection: (or <Enter> to exit) 10

===========================================
|| Welcome to STARMA analyzing interpreter ||
===========================================

# quit

See you!

[IEAST v1.30.01 - STARMA Modeling & Analysis]

================ Main Menu ================
  [ 1] Setup
  [ 2] Data Preprocessing
  [ 3] AutoCorrelation Analysis
  [ 4] Model Identification
  [ 5] Parameter Estimation
  [ 6] Diagnostic Analysis
  [ 7] Forecasting
  [ 8] Simulation
  [ 9] Preference
  [10] Interpreter
  [11] Exit
===========================================

Selection: (or <Enter> to exit)
```

10.1.2 Program Mode and Interactive Mode

There are two modes in the interpreter, i.e. *program mode* and *interactive mode*. Under the prompt '`#`', if the first word of the inputs is a command (listed in section 10.3), this line of statement will be treated in an interactive way and executed right away. If the first word is a number (a line number separated by tailing space), this line of input will be put into system memory as a part of a program and not be executed in the mean time. Simply speaking, in the interpreter prompt input line, if the leading is a integer number (from 1 to 10000), then the entire line will be put into system memory as part of a stand-by program. Otherwise, the command entered will be executed right after the `Enter` key pressed.

Take a look at the example below. The input "`list`" after prompt '`#`' means show current program. But, after that, the input "`45 list`" means store the statement "list" into line 45 of the program. When we list the program again, there is an extra line (`45 list`) added.

```
# list
10 load data verification.dat
20 load weight uniform.wet
30 stacf ST_ACF Z 16 3
40 plot ST_ACF 16 3 "ACF"
50 end
#
# 45 list
#
# list
10 load data verification.dat
```

```
20 load weight uniform.wet
30 stacf ST_ACF Z 16 3
40 plot ST_ACF 16 3 "ACF"
45 list
50 end
#
```

10.1.3 Notices for the Interpreter

- All data types, data structures, variables, expressions, functions, and operators are inherited from of GNU Octave (http://www.octave.org).
- Line number is for program sequence reference and flow control. Line number can be a integer from 1 to 10000. Line number 0 is reserved for system use. In program, the line number can be arbitrarily chosen, however, the line numbers are the running sequence of the program. Users need to maintain its logical order in the sense of line numbers.
- The executing sequence will follow the order of the *line numbers* (from small to big) in a program.
- The newly entered line of program will overwrite the existing one with the same line number in memory.
- All programs need to be either loaded or entered into system memory first, then type command 'run' to run the program.
- Spaces are only used for separation among commands and arguments.
- Any line of a program should begin with a line number, then followed by command and arguments as shown below.

  ```
  # list
  10 load data verification.dat
  20 load weight uniform.wet
  30 stacf ST_ACF Z 16 3
  40 plot ST_ACF 16 3 "ACF"
  50 end
  #
  ```

- All global variables are shared between menu-driven and interpreter. If one is changed under interpreter, "the one" in menu-driven is changed as well.

10.2 A Sample Program

In this section, to illustrate how to use the interpreter, a dataset which was generated by simulation according to a given (known) model was analyzed and modeled. Using IEAST, we can find an appropriate model (type and orders) to represent the dataset. Furthermore, some diagnoses were provided for model adequacy evaluation. Finally, the acquired model and parameters were compared with the given model. In this example, we found that it is efficient and flexible to analyze a given dataset with the programming environment of IEAST.

10.2.1 Dataset Generation

First, we needed to generate a dataset as the input by simulating a given STARMA model with known parameters. The dataset for this example was generated based on the following specifications:

- Spatial dimensions: two-dimensional space, 8×8 sites.
- Model type and orders: STARMA(mixed model) with both maximum temporal and spatial orders set equal to one, that is, STARMA(1,1).
- Mathematical expression of the model:

$$\begin{aligned} \mathbf{Z}(t) &= (0.5) * \mathbf{Z}(t-1) + (-0.1) * \mathbf{W}^{(1)}\mathbf{Z}(t-1) + \varepsilon(t) \\ &\quad -(-0.3) * \varepsilon(t-1) - (0.07) * \mathbf{W}^{(1)}\varepsilon(t-1). \end{aligned}$$

- Random noise: $\varepsilon(t) \sim N(\boldsymbol{\mu} = \mathbf{0}, \boldsymbol{\sigma}^2\mathbf{I} = 0.00125\mathbf{I})$.
- Weight matrices: Uniform, filename: "uniform.wet".
- Data set: 100 points in time, 64 sites in two dimensional space, filename: "demo.dat". The dataset were generated by picking the last 100 generations from 5000 simulated generations.

10.2.2 Modeling Procedures

As mentioned, model types and orders of some models can be simply recognized by investigating their STACF and STPACF, especially for pure STAR and pure STMA models. In such cases, after STACF / STPACF have been calculated, the model type and orders of the model can be decided immediately. However, in reality, there exist many models, especially mixed models i.e. STARMA, which are not easily recognized by investigating their STACF / STPACF behaviors. Hence, a feasible way to discover the model type is to test AICC and BIC for every possible model type with maximum possible orders iteratively and choose the best one. Then, by evaluating significance of every parameter in the candidate and dropping the insignificant parameters, an appropriate and parsimonious model can be obtained.

In the identification stage, the STACF and STPACF of the dataset were computed. If their behaviors are obvious as expected, the model type and orders of the candidate can be decided immediately. However, in this STARMA example, it was not the case (see the autocorrelation analysis in Figure 10.1). All the models of three different types (i.e. STAR, STMA, STARMA) with the largest possible orders (in time and space) were chosen to be candidates.

According to the model types of the candidates selected in identification stage, in the pre-estimation stage, initial values for the estimation stage were estimated. Then, these initial values were fed into the estimation stage. During the optimization process of estimation stage, these candidate models were fitted to the dataset, iteratively, and the estimates of parameters were found.

When the model fitting was finished, the program entered a diagnosis stage. There were diagnostic criteria generated (AICC, BIC, residual variance and parameter significance). Based on the values of these criteria, the appropriate model type and orders can be decided by user inspection or program selection. If necessary, the whole process can be re-iterated and the type and orders adjusted for the next run according to the criteria obtained.

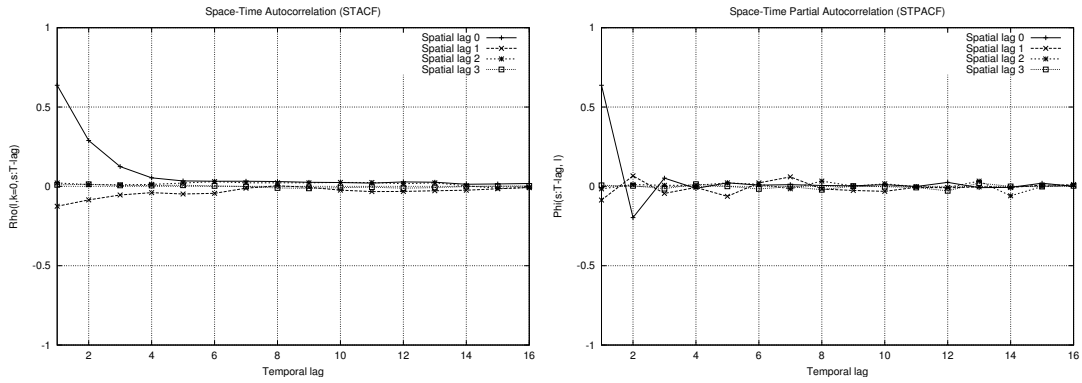

Figure 10.1: STACF and STPACF for the simulated dataset STARMA(1,1)

10.2.3 Program Description

To iteratively test all three possible models (STAR, STMA, STARMA), there is a *for* loop from line 100 to 250 which was used to control the program flow. Dataset and uniform weight matrices were loaded to **Z**, **W** in line 10 and 20, respectively. Line 30-60 calculated and plotted STACF and STPACF. Line 80 assigned the highest possible temporal and spatial orders. Line 150 was for pre-estimating the initial values of the parameters using either space-time extension of Yule-Walker or Hannan-Rissanen algorithm according to the model type. If the model is STAR, the Yule-Walker equation is used. Otherwise, space-time extension of Hannan-Rissanen algorithm is used. Line 160 was for estimating parameters based on the initial values acquired in last steps. Lines 180 and 190 calculated the AICC and BIC values for every candidate, respectively. Line 200 calculated the variance of the residuals. After the *for* loop, the candidate with lowest AICC/BIC was chosen to be the final model type. Then, in line 310, the significance of parameters was calculated, and the necessary parameters decided.

In this program, the words in lowercase were commands, and all variables were in capital. The meaning of variables are described below. ST_ACF and ST_PACF were used to store the space-time autocorrelation results. PHI and THETA were matrices which stored the coefficients of autoregressive terms and moving average terms, respectively. Z, W, MODEL, MAX_T_ORDER and MAX_S_ORDER are system-reserved global variables which represent the dataset($\mathbf{Z}(t)$), the weight matrices($\mathbf{W}^{(l)}$), the model type, and the maximum orders in time or space, respectively. The leading 'f' of some variables meant final results, the leading 'e' meant estimated results, the leading 'i' meant initial values, the leading 's' meant statistical significance. The variable MASK was a matrix used to disable some unimportant or not interesting parameters (in PHI or THETA) by simply setting to zero to disable or setting to one to enable on the corresponding positions of matrices. In this example, all parameters in either PHI or THETA were enabled.

10.2.4 Results

The program and outputs of this example modeling procedure are shown below. After entering the interpreter in IEAST (by choosing selection 10 in the main menu of menu-driven mode), there came an interactive prompt for commands. Use *load* command to retrieve the modeling procedure program in file *"demo.pgm"*. The first part of the following results are the program listing. The two tables following the program are the sample STACF and STPACF of the dataset. The results

followed the tables are analysis output of pre-estimation/estimation and diagnostic checking.

```
==========================================
|| Welcome to STARMA analyzing interpreter ||
==========================================

# load program demo.pgm
# list
 10 load data demo.dat
 20 load weight uniform.wet
 30 stacf ST_ACF Z 16 3
 40 plotacf ST_ACF 16 3 "ACF"
 50 stpacf ST_PACF Z 16 3
 60 plotacf ST_PACF 16 3 "PACF"
 70 let fAICC=0 MODEL_MAX_T=1 MODEL_MAX_S=1 MASK=ones(MODEL_MAX_T,MODEL_MAX_S+1)
 80 let types=["STAR";"STMA";"ARMA"]
 90 for I 1 1 3
100 let MODEL=types(I,:)
110 print string $\n==============$
120 print var MODEL
130 print string $==============$
140 let iTHETA=zeros(size(MASK)) iPHI=zeros(size(MASK))
150 pre_estimate iPHI iTHETA MODEL MODEL_MAX_T MODEL_MAX_T MODEL_MAX_S
160 estimate ePHI eTHETA eRESIDUALS iPHI iTHETA MASK
170 print model MODEL ePHI eTHETA
180 aicc AICC ePHI eTHETA eRESIDUALS
190 bic BIC ePHI eTHETA eRESIDUALS
200 variance eVARIANCE eRESIDUALS
210 print string $[ AICC/BIC/Variance ]\n$
220 print var AICC BIC eVARIANCE
230 if fAICC<AICC #250
240 let fAICC=AICC fBIC=BIC fTYPE=MODEL fPHI=ePHI fTHETA=eTHETA fRESIDUALS=eRESIDUALS fVARIANCE=eVARIANCE
250 endfor
260 print string $\n***************\n* Final Model *\n***************\n$
270 let MODEL=fTYPE
280 print var MODEL fAICC fBIC fVARIANCE
290 print model MODEL fPHI fTHETA
300 print string $\n[ Significance ]\n$
310 significance sPHI sTHETA ePHI eTHETA MASK
320 print var sPHI sTHETA
330 print string $\n[ Residual's STACF/STPACF analysis ]\n$
340 stacf ACF_RESIDUALS eRESIDUALS 16 3
350 plotacf ACF_RESIDUALS 16 3 "ACF"
360 stpacf PACF_RESIDUALS eRESIDUALS 16 3
370 plotacf PACF_RESIDUALS 16 3 "PACF"
380 end
#
# run

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Autocorrelation of Z, Rho(s, l, k=0)
-----------------------------
 S-lag  0     1     2     3
T-lag------------------------
  1   .661 -.152  .026  .038
```

```
  2   .334 -.110  .025  .026
  3   .159 -.070  .023  .010
  4   .068 -.048  .015  .003
  5   .022 -.033  .013  .001
  6   .004 -.026  .004  .002
  7   .001 -.023  .001  .000
  8   .005 -.017  .007 -.005
  9  -.003 -.005  .007  .000
 10  -.011  .008  .003  .003
 11  -.010  .006  .009  .005
 12  -.006 -.001  .013  .009
 13   .008 -.014  .007  .018
 14   .009 -.023  .002  .010
 15   .004 -.020 -.001  .009
 16   .001 -.011  .001 -.002

Calculating STPACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Partial Autocorrelation of Z, Phi(s,l)
-----------------------------
 S-lag  0     1     2     3
T-lag------------------------
  1   .661 -.099  .003  .011
  2  -.185  .081 -.009 -.017
  3   .037 -.031  .015 -.016
  4  -.020 -.014 -.014  .011
  5  -.002 -.003  .012 -.007
  6   .000 -.019 -.026  .008
  7   .005 -.014  .008 -.011
  8   .005  .008  .015 -.008
  9  -.017  .022 -.008  .020
 10  -.004  .015 -.001 -.001
 11   .005 -.033  .033  .011
 12   .000  .005 -.005  .005
 13   .020 -.038 -.016  .022
 14  -.013 -.001  .001 -.030
 15  -.000  .001 -.003  .024
 16  -.001  .007  .013 -.033

==============
MODEL = STAR
==============

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

------- Iter#1:(Mon Aug 09 22:18:39 2004) -------

ePHI =

   0.65  -0.10

Advancement of Average Parameter SSE = 0.000003
Time elapsed = 0:0:3.95

Z(t) =
( 0.65)*Z(t-1) + (-0.10)*W1*Z(t-1) +
e(t)
```

```
[ AICC/BIC/Variance ]

AICC =  -6.6371e+04
BIC =  -2.8178e+04
eVARIANCE = 0.0013105

==============
MODEL = STMA
==============

------- Iter#1:(Mon Aug 09 22:18:48 2004) -------

eTHETA =

  -0.29   0.08

Advancement of Average Parameter SSE = 0.164246
Time elapsed = 0:0:5.39

------- Iter#2:(Mon Aug 09 22:18:53 2004) -------

eTHETA =

  -0.55   0.13

Advancement of Average Parameter SSE = 0.001766
Time elapsed = 0:0:6.11

------- Iter#3:(Mon Aug 09 22:18:59 2004) -------

eTHETA =

  -0.60   0.11

Advancement of Average Parameter SSE = 0.000081
Time elapsed = 0:0:4.54

------- Iter#4:(Mon Aug 09 22:19:04 2004) -------

eTHETA =

  -0.61   0.10

Advancement of Average Parameter SSE = 0.000010
Time elapsed = 0:0:4.27

Z(t) = e(t)
- (-0.62)*e(t-1) - ( 0.10)*W1*e(t-1)

[ AICC/BIC/Variance ]

AICC =  -6.5541e+04
BIC =  -2.7763e+04
eVARIANCE = 0.0014238

==============
MODEL = ARMA
==============

------- Iter#1:(Mon Aug 09 22:19:13 2004) -------

ePHI =
```

```
   0.50  -0.10

eTHETA =

  -0.29   0.08

Advancement of Average Parameter SSE = 0.000015
Time elapsed = 0:0:18.33

Z(t) =
( 0.50)*Z(t-1) + (-0.10)*W(1)*Z(t-1) +
e(t)
 - (-0.29)*e(t-1) - ( 0.08)*W(1)*e(t-1)

[ AICC/BIC/Variance ]

AICC =  -6.6759e+04
BIC =  -2.8363e+04
eVARIANCE = 0.0012603

***************
* Final Model *
***************

MODEL = ARMA
fAICC =  -6.6759e+04
fBIC =  -2.8363e+04
fVARIANCE = 0.0012603
Z(t) =
( 0.50)*Z(t-1) + (-0.10)*W(1)*Z(t-1) +
e(t)
 - (-0.29)*e(t-1) - ( 0.08)*W(1)*e(t-1)

[ Significance ]

sPHI =

  0.0010000  0.0010000

sTHETA =

  0.0010000  0.0100000

[ Residual's STACF/STPACF analysis ]

Initializing Covariance Look-up Table CovarTB[] for major varialbe Z....
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Calculating STACF...
|----------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Autocorrelation of Z, Rho(s, l, k=0)
-----------------------------
 S-lag  0     1     2     3
T-lag------------------------
  1   .002 -.001  .010  .012
  2   .006 -.001 -.002  .011
  3  -.007  .011  .008 -.013
  4  -.008 -.006 -.008  .001
```

```
  5  -.013 -.000  .011 -.003
  6  -.011 -.002 -.005  .001
  7  -.007 -.010 -.009  .004
  8   .012 -.008  .007 -.013
  9   .001 -.009  .010  .006
 10  -.015  .023 -.010 -.000
 11   .002 -.003  .005  .003
 12  -.016  .011  .015 -.006
 13   .016 -.010 -.001  .022
 14   .002 -.013  .001 -.007
 15  -.001 -.006 -.007  .014
 16  -.005 -.007  .006  .004

Calculating STPACF...
|--------------------------------------|
|>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>|

Space-Time Partial Autocorrelation of Z, Phi(s,l)
-----------------------------
 S-lag  0     1     2     3
T-lag-------------------------
  1   .002 -.003  .016  .021
  2   .006 -.001 -.004  .019
  3  -.007  .020  .013 -.022
  4  -.008 -.013 -.014  .001
  5  -.013 -.001  .018 -.005
  6  -.011 -.003 -.009  .001
  7  -.007 -.019 -.015  .007
  8   .012 -.015  .013 -.023
  9   .001 -.016  .016  .010
 10  -.016  .044 -.018 -.002
 11   .002 -.008  .010  .008
 12  -.016  .019  .026 -.011
 13   .016 -.018 -.003  .036
 14   .002 -.025  .000 -.013
 15  -.001 -.011 -.011  .024
 16  -.005 -.012  .009  .008
#
```

10.3 Command References

In this section, a comprehensive description of all instructions in the interpreter is given.

The command keywords are in **bold face**. In *Syntax*, the arguments in "`< >`" are required; arguments in "`[ ]`" are optional; symbol "`|`" means "OR". For example, "`command1 < arg1 | arg2 >`" means command1 has one required argument which is either arg1 or arg2. On other hand, "`command1 [ arg1 | arg2 ]`" means command1 can have none or one argument which is either arg1 or arg2.

All the instructions (IEAST version 1.30.01) and their usage are listed below.

aicc

Synopsis: **aicc** `<result> <final_Phi> <final_Theta> <final_EE>`
Description: This instruction is used to calculate bias-corrected Akaie's information criterion (AICC). The result is a real number. Three arguments are used as inputs which are estimated AR coefficients `final_Phi`, estimated MA coefficients `final_Theta` and final residuals `fianl_EE`.

Example:

```
# aicc AICC EST_PHI EST_THETA EST_EE
```

autoiden

Synopsis: **autoiden** <result_type> <result_maxs> <result_maxt> <stacf> <stpacf>
Description: This instruction is to automatically identify the model type and maximum spatial/temporal orders based on the analyses of space-time autocorrelations. The input arguments are <stacf> and <stpacf> which are the matrices of the space-time autocorrelation and partial autocorrelation. The output will be put into three variables: <result_type>, <result_maxs>, and <result_maxt>. The model type is in <result_type> in string type ('STAR','STMA', or 'ARMA'). The maximum spatial order is in <result_maxs>. The maximum temporal order is in <result_maxt>.
Example:

```
# load data testdata.dat
# stacf ACF Z 16 4
# stpacf PACF Z 16 4
# autoiden TYPE MAXS MAXT ACF PACF
#
```

bic

Synopsis: **bic** <result> <final_Phi> <final_Theta> <final_EE>
Description: This instruction is used to calculate Bayesian information criterion (BIC) which is used to correct some tendency in AICC testing. The result is a real number. Three arguments are used as inputs which are estimated AR coefficients final_Phi, estimated MA coefficients final_Theta and final residuals fianl_EE.
Example:

```
# bic BIC EST_PHI EST_THETA EST_EE
```

calctrend

Synopsis: **calctrend** <result> <Z> <S|T> <method> [arg]
Description: Calculate spatial or temporal trends of dataset <Z>. Result, the trend, will be stored in the variable specified in <result>. <Z> is the dataset to be calculated. Third argument <S|T> specifies either spatial(S) or temporal(T) trend is going to be found. ¡method¿ specifies the method used to find the trends, there are three different options: POLYREG for polynomial regression, FOURIER for Fourier transformation, and AVERAGE for averaging over time. The last argument [arg] is used by POLYREG and FOURIER. In POLYREG, it means the maximum order of the polynomial used to fit the dataset. In FOURIER, means the highest harmonic in spectrum for the trend.
Example: calculate third order polynomial regression spatial trend surface and save to the variable 'STREND'.

```
# calctrend STREND Z S POLYREG 3
```

clearmode

Synopsis: **clearmode** [TRACE | HIDEACF | PLOTEPS]
Description: Clear the environmental settings.

dec

Synopsis: **dec** `<var_name>`
Description: Decrease the value of the scalar variable `var_name` by one.

demean

Synopsis: **demean**
Description: Remove the mean of current series in the global variable Z.

detrend

Synopsis: **detrend** `<result> <Z> <Strend> <Ttrend> <method> [arg]`
Description: According to the spatial and temporal trends found, there are several ways to detrend them from the dataset `<Z>`. They are (1)detrend spatial trend only (method=ALL_S); (2)detrend temporal trend only (method=ALL_T); (3)detrend the best-fit (to the dataset) spatial trend (method=BF_SS); (4)detrend the dataset by the product of temporal trend and spatial trend (method=SMULT); (5) detrend the best-fit Strend*Ttrend from the dataset (method=BF_ST). Result will be stored in the variable specified in `<result>`. `<Z>` is the dataset to be detrended. `<Strend>` and `<Ttrend>` are the trends have been found. `[arg]` is for additional argument, not used yet.
Example: detrend both spatial and temporal trends from the dataset Z by the method BF_ST. The spatial trend and temporal trend have been found and save in the variables STREND and TTREND, respectively. The result will be saved into the variable DTZ.

```
# detrend DTZ Z STREND TTREND BF_ST
```

diff

Synopsis: **diff** `Z T <Z> <n>`
Description: Differencing the current series by `n`, i.e. `(1-B^n)Z`.
Example: De-seasonalize the current series.

```
# diff Z T <Z> 12
```

dir

Synopsis: **dir** `[ data | weight | program ]`
Description: List the files in current directory. If the optional argument is `data` then list dataset files only. If `weight`, then weighting files only. If `program`, then program files only. If no argument, list them all.
Example: list all the files in current directory.

```
# dir
```

Example: list all weight matrix files in current directory.

```
# dir weight
```

end

Synopsis: **end**
Description: Terminate the program and back to prompt "`#`".

estimate

Synopsis: **estimate** `<est_phi> <est_theta> <final_EE> <init_phi> <init_theta> [AR_Mask] [MA_Mask]`

Description: Using the initial values as start, the optimization process iteratively approaches to the final estimated parameters. This optimization process is based on Marquardt's method. In arguments, the initial values as used as inputs in `init_phi` and `init_theta`. The final estimated parameters are stored into the variables `est_phi` and `est_theta`. The final residuals are in `final_EE`. `AR_Mask` and `MA_Mask` are used to decide which parameters should be taking into account during optimization process. Users can disable or enable any single parameter in the parameter matrices `est_phi` or `est_theta`. To enable/disable the specific parameter, simply set the corresponding element in matrix `AR_Mask` or `MA_Mask` to one/zero.
Example:

```
# estimate EST_PHI EST_THETA EST_EE INIT_PHI INIT_THETA AR_MASK MA_MASK
```

estimate_hr

Synopsis: **estimate_hr** `<result_phi> <result_theta> <AR_max_t> <MA_max_t> <max_s_order>`
Description: Find the initial values for the coming estimation process using Hannan-Rissanen algorithm. It is good for non-linear processes such as STMA or STARMA models. In arguments, the maximum temporal order in AR terms is `AR_max_t`, the maximum temporal order in MA terms is `MA_max_t`, and the maximum spatial order is `max_s_order`. The final estimated initial values are stored into the variables `result_phi`, `result_theta`.
Example:

```
# estimate_hr INIT_PHI INIT_THETA 2 2 1
```

estimate_yw

Synopsis: **estimate_yw** `<result_phi> <AR_max_t> <max_s_order>`
Description: Find the initial values for the coming estimation process using Yule-Walker equations. It is good for linear processes such as STAR models. In arguments, the maximum temporal order in AR terms is `AR_max_t` and the maximum spatial order is assigned in `max_s_order`. The final estimated initial value are stored into the variables `result_phi`.
Example:

```
# estimate_yw INIT_PHI 2 1
```

exit

Same as the instruction **quit**.

for ... endfor

Synopsis: **for** `<var> <init> <step> <final>` **endfor**
Description: This is a for-loop instruction. The variable `var` will start from `init` and decrease or increase according to the value in `step` for each iteration until reaching `final`.
Example: Let the variable `i` count down from `10` to `-10` with step `-1`.

```
40 for i 10 -1 -10
...
...
80 endfor
```

goto

Synopsis: **goto** `<#line_no>`
Description: Change the flow of the program to `#line_no`.
Example:

```
80 let A=A+1
...
100 goto #80
```

help

Synopsis: **help**
Description: Show help menu for commands.

if

Synopsis: **if** `<condition> <#line_no>`
Description: Conditionally changig the program flow. If the value of the condition is not zero, then goto line `#line_no`. Otherwise, do nothing and continue on next line. Condition must be a valid arithmetic statement.
Example:

```
60 if fAICC<AICC #90
70 let TYPE='STAR'
80 fAICC=AICC
90 let TYPE='STMA'
```

inc

Synopsis: **inc** `<var_name>`
Description: Increase the value of the scalar variable `var_name` by one.

let

Synopsis: **let** `<assignment> [assignment] [assignment]`
Description: Assign one or multiple variable values. `<assignment>` must be a valid Octave assignment statement.
Example:

```
# let MATRIX_A=[1,3;2,1] B=3 MATRIX_C=inv(MATRIX_A)*B
```

list

Synopsis: **list** `[#begin_line] [#end_line]`
Description: Show all or part of a program in memory from #begin_line to #end_line.
Example:

```
# list 10 100
```

load

Synopsis: **load** `< data | program | weight > <filename>`
Description: Load data, program or weight matrices into memory. The valid file name for data is '*.dat', program is '*.pgm', and weight matrices is '*.wet'.
Example: load in the program file 'verify.pgm'

```
# load program verify.pgm
```

plotacf

Synopsis: **plotacf** `<acf_matrix> <max_t_lag> <max_s_lag>`
Description: Plot the space-time autocorrelation matrices.
Example:

```
# plotacf STACF 16 3
```

plot2d

Synopsis: **plot2d** <2d.data> <var.caption> [<xlabel> <ylabel>]
Description: Plot the array <2d.data> in a two-dimensional graph with caption and X/Y labels. X is the the index of the array. Y is the values of the corresponding elements.
Example:

```
# plot2d TTrendZ $Temporal Trend$
```

plot3d

Synopsis: **plot3d** <3d.data> <mode> <var.caption>
Description: Plot the matrix <2d.data> in a three-dimensional graph with caption. <mode> is the style of the 3D plot, and can be S, M or B. S for surface only. M for mosaic map style. B for Both.
Example:

```
# plot3d STrendZ M $Spatial Trend$
```

pre_estimate

Synopsis: **pre_estimate** <init_phi> <init_theta> <model> <AR_max_t> <MA_max_t> <max_s_order>
Description: This instruction find the initial values of parameters using different algorithms according to their model types.
Example:

```
# pre_estimate INIT_PHI INIT_THETA MODEL 2 2 1
```

print

Synopsis: **print** <model|var|setup|string|acf> [<type> <PHI|THE|PHI THE> | <variable1> [<variable2>]... | $Var.String$ | <Var.ACF> [<Max.T.Lag> [Max.S.Lag]]]
Description: Print model equation, variable values, system cinfigurations, a string (enclosed by $...$), or autocorrelations.
Example: To print a model equation with type in the variable 'TYPE', model AR parameters in 'MY_PHI', and model MA parameters in 'MY_THE'. Then, to print the variable 'VARIANCE' and the autocorrelation result in the variable 'STACF'.

```
# print model TYPE MY_PHI MY_THE

Z(t) =
( 0.26)*Z(t-1) + ( 0.37)*W1*Z(t-1) + ( 0.10)*W2*Z(t-1) + (-0.09)*W3*Z(t-1) +
( 0.04)*Z(t-2) + (-0.18)*W1*Z(t-2) + (-0.09)*W2*Z(t-2) + (-0.06)*W3*Z(t-2) +
(-0.02)*Z(t-3) + (-0.10)*W1*Z(t-3) + ( 0.00)*W2*Z(t-3) + (-0.04)*W3*Z(t-3) +
e(t)

# print var VARIANCE

 VARIANCE = 0.0012665

# print acf STACF 16 4
-----------------------------------------
```

```
   S-lag    0      1      2      3      4
T-lag  ------------------------------------
   1     0.332  0.280  0.163 -0.006  0.064
   2     0.110  0.009  0.007 -0.065 -0.050
   3    -0.034 -0.102 -0.082 -0.104 -0.093
   4    -0.044 -0.093 -0.087 -0.078 -0.078
   5    -0.032 -0.063 -0.041 -0.032 -0.052
   6    -0.013 -0.035 -0.040 -0.036 -0.040
   7    -0.014 -0.010 -0.014 -0.020 -0.028
   8    -0.004 -0.003 -0.014 -0.015 -0.005
   9     0.000 -0.003 -0.003  0.003 -0.004
  10     0.008 -0.013 -0.006 -0.013 -0.004
  11    -0.003 -0.007  0.003 -0.006 -0.002
  12     0.007 -0.006 -0.008 -0.007 -0.002
  13     0.004  0.005  0.006 -0.004 -0.001
  14     0.011  0.011  0.008 -0.001 -0.001
  15     0.004  0.006  0.004  0.005 -0.001
  16     0.002  0.001 -0.002  0.002 -0.002
```

quit

Synopsis: **quit**
Description: Leave the interpreter, and back to menu-driven environment.

rem

Synopsis: **rem** `<any string for remark>`
Description: Comment lines, any line with the 'rem' as its beginning whill be skipped while running.
Example:

```
# rem the following calculate STPACF
```

renumber

Synopsis: **renumber**
Description: Rearrange all line numbers, start from 10, increase by 10. The line numbers used by flow control commands are adjusted according to their relative locations. In other words, the line number may change accordingly.

run

Synopsis: **run**
Description: Execute the program loaded from the first line of the program.

save

Synopsis: **save** `< data | program | weight > <filename>`
Description: Save data, IEAST program or weight matrices from memory to files. The valid file name for data is '*.dat', for program is '*.pgm' and for weight matrices is '*.wet'.
Example: save the data to the data file 'demo.dat'.

```
# save data demo.pgm
```

setmode

Synopsis: **setmode** `[ TRACE | HIDEACF | PLOTEPS ]`
Description: Set the environmental settings.
Example: print the line number while running for debugging purpose.

```
# setmode TRACE
```

Example: don't print the results of stacf or stpacf while running, only calculating.

```
# setmode HIDEACF
```

Example: not only output the graphics with GNUPLOT but also generate a corresponding eps file.

```
# setmode PLOTEPS
```

shell

Synopsis: **shell** `<command>`
Description: Execute a shell command under IEAST interpreter.

significance

Synopsis: **significance** `<sig_Phi> <sig_Theta> <final_Phi> <final_Theta> [AR_Mask] [MA_Mask]`
Description: Statistical significance is to evaluate the importance of the parameters obtained from estimation stage. The outputs of this instruction are stored in `sig_Phi` and `sig_Theta`, which are the same size as the estimated parameters `final_Phi` and `final_Theta`. The elements in `sig_Phi` and `sig_Theta` represent the importance probability (from $0.0\tilde{1}.0$). The bigger the value is, the higher the importance. `AR_Mask` and `MA_Mask` has the same effect as in the instruction estimate.
Example:

```
# significance sig_PHI sig_THETA EST_PHI EST_THETA
```

simulate

Synopsis: **simulate** `<result> <sim_phi> <sim_theta> <#iteration> <#generation> <noise_var> <noise_mean>`
Description: This instruction is doing simulation according to the assigned specifications, which are described as below. `sim_phi` is the coefficients of AR terms in matrix form. `sim_theta` is the coefficients of MA terms in matrix form. The simulated results are picked up after the number of generations assigned in `#iteration`. The number of generation simulated is specified in `#generation`. `noise_var` and `noise_mean` represent the statistical characteristics of the random disturbance.
Example:

```
# simulate STAR-T2S1N100 sPHI sTHETA 5000 100 0.00125 0.0
```

stacf

Synopsis: **stacf** `<st_acf> <Z> <max_t_lag> <max_s_lag>`
Description: Calculate space-time autocorrelation according to the series data in series `Z`, the maximum time lag `max_t_lag`, and the maximum spatial lag `max_s_lag`. The final result is stored into the matrix variable `stacf`.
Example:

```
# stacf ST_ACF Z 16 3
```

stop

Synopsis: **stop**
Description: Temporarily pause the program for program diagnosis or peeking analysis results. The program will continue when an empty line is entered (press `Enter` right after the prompt '`#`'). Every command *GNU Octave* can accept can be used in *stop*.

stpacf

Synopsis: **stpacf** `<st_pacf> <Z> <max_t_lag> <max_s_lag>`
Description: Calculate partial space-time autocorrelation according to the data in series `Z`, the maximum time lag `max_t_lag`, and the maximum spatial lag `max_s_lag`. The final result is stored into the matrix variable `stpacf`.
Example:

```
# stpacf ST_PACF Z 16 3
```

stxcf

Synopsis: **stxcf** `<result> <Var1> <Var2> <max_t_lag> <max_s_lag>`
Description: Calculate space-time cross correlation between two space-time variables ¡Var1¿ and ¡Var2¿ with the maximum time lag `max_t_lag` and the maximum spatial lag `max_s_lag`. The final result is stored into the matrix variable `<result>`.
Example:

```
# load data WNV_XHuman_CR45_10x10.dat
# load data WNV_XCrow_CR45_10x10.dat Y
# stxcf STXCF Z Y 20 4
```

stxcf_ext

Synopsis: **stxcf_ext** `<result> <Var1> <Var2> <max_t_lag> <max_s_lag>`
Description: Calculate space-time cross correlation between two space-time variables `<Var1>` and `<Var2>` with maximum spatial lag `max_s_lag` from temporal lag `-max_t_lag` to `max_t_lag`. The final result is stored into the matrix variable `<result>`.
Example:

```
# load data WNV_XHuman_CR45_10x10.dat
# load data WNV_XCrow_CR45_10x10.dat Y
# stxcf_ext ExtSTXCF Z Y 20 4
```

variance

Synopsis: **variance** `<variance> <residuals>`
Description: The instruction variance is to calculate the variance of the residuals. There are one input variable `residuals` and one output `variance` in its format. The final result (variance) will be stored in the variable `variance`.
Example:

```
# variance EST_VARIANCE EST_EE
```

Bibliography

[1] Akaike, H. (1973) Information theory and an extension of the maximum likelihood principle. In: Petrov, B., F., C. (Eds.) *2nd International Symposium on Information Theory*, Akademiai Kiado, Budapest, 267–281.

[2] Akaike, H. (1978) Time series analysis and control through parametric models. In: Findley, D. F. (Ed.), *Applied Time Series Analysis*, Academic Press, New York, 1–23.

[3] Box, G.E.P., Jenkins, G.M. (1970) *Time Series Analysis: Forecasting and Control.* Holden-Day, San Francisco.

[4] Hannan, E.J., Rissanen, J. (1982) Recursive estimation of mixed auto-regressive moving average order. *Biometrika*, 69, 81–94.

[5] Hordijk, L., Nijkamp, P. (1977) Dynamic models of spatial autocorrelation. *Environment and Planning A*, 9, 505–519.

[6] Hurvich, C.M., Tsai, C.-L. (1989) Regression and time series model selection in small samples. *Biometrika*, 76, 297–307.

[7] Jones, R.H. (1975) Fitting autoregressions. *Journal of the American Statistical Association*, 70 (351), 590–592.

[8] Marquardt, D.W. (1963) An algorithm for least squares estimation of nonlinear parameters. *Journal of the Society of Industrial And Applied Mathematics*, 11, 431–441.

[9] Martin, R.L. and Oeppen, J.E. (1975) The identification of regional forecasting models using space-time correlation functions. *Transactions of the Institute of British Geographers*, 66, 95–118.

[10] Pfeifer, P.E. (1979) *Spatial-Dynamic Modeling.* Unpublished Ph.D. dissertation, Georgia Institute of Technology, Atlanta, Georgia.

[11] Pfeifer, P.E., Deutsch, S.J. (1980) Stationarity and invertibility regions for low order STARMA models. *Communications in Statistics - Simulation and Computation*, 9 (5), 551–562.

[12] Pfeifer, P.E., Deutsch, S.J. (1980) A three-stage iterative procedure for space-time modeling. *Technometrics*, 22 (1), 35–47.

[13] Pfeifer, P.E., Deutsch, S.J. (1981) Variance of the sample space-time autocorrelation function. *Journal of the Royal Statistical Society Series B*, 43 (1), 28–33.

[14] Schwarz, G. (1978) Estimating the dimension of a model. *The Annals of Statistics*, 6 (2), 461–464.

[15] Shibata, R. (1976) Selection of the order of an autoregressive model by Akaike's information criterion. *Biometrika* 63, 117–126.

[16] Stoffer, D.S. (1986) Modeling risk from a disease in time and space. *Journal of the American Statistical Association*, 81 (395), 762–772.